



Tiago Miguel Augusto Ferreira

Licenciado em Ciências da
Engenharia Eletrotécnica e de Computadores

Estudo do processamento de simulações da evolução de incêndios florestais em ambiente CLOUD

Dissertação para obtenção de Grau Mestre em
Engenharia Eletrotécnica e de Computadores

Orientador: Rui Santos-Tavares, Professor Auxiliar,
Universidade Nova de Lisboa
Faculdade de Ciências e Tecnologias



FACULDADE DE
CIÊNCIAS E TECNOLOGIA
UNIVERSIDADE NOVA DE LISBOA

Setembro, 2019

**Estudo do processamento de simulações de incêndios florestais em ambiente
*CLOUD***

Copyright © Tiago Miguel Augusto Ferreira, Faculdade de Ciências e Tecnologia,
Universidade Nova de Lisboa.

A Faculdade de Ciências e Tecnologia e a Universidade Nova de Lisboa têm o direito, perpétuo e sem limites geográficos, de arquivar e *publicar* esta dissertação através de exemplares impressos reproduzidos em papel ou de forma digital, ou por qualquer outro meio conhecido ou que venha a ser inventado, e de a divulgar através de repositórios científicos e de admitir a sua cópia e distribuição com objectivos educacionais ou de investigação, não comerciais, desde que seja dado crédito ao autor e editor.

Para os meus Familiares

Agradecimentos

Gostaria de agradecer a todos aqueles que contribuíram para a realização desta dissertação.

Em primeiro lugar quero agradecer ao projecto foRESTER pela possibilidade de aplicação dos resultados obtidos e os recursos disponibilizados pelo projecto.

Queria agradecer ao meu orientador Professor Rui Santos-Tavares pela oportunidade de realizar e desenvolver este tema de dissertação.

Aos meus amigos de curso Ricardo Domingues, André Teixeira, Fábio Ribeiro, Sérgio Brígida, Pedro Varandas, Pedro Garcia, João Neves, Hernâni Domingos e Abraham Guerra, obrigado a todos pelo apoio que deram ao longo deste meu percurso académico. Vocês são excelentes amigos e estiveram sempre presentes nos bons e maus momentos.

Gostaria de agradecer também aos meus amigos, Cláudio Quaresma, Ricardo Anjos, Filipe Oliveira, Alexandre Pereira, Eduardo Cartaxana, Eduardo Brandão, Cátia Moura, Joana Amorim e Jéssica Caeiro, por toda a força que me deram ao longo deste percurso académico e desenvolvimento desta dissertação.

Um especial e forte agradecimento à minha namorada Daniela Fasero por todo o apoio incondicional, motivação e paciência ao longo do desenvolvimento desta dissertação e percurso académico. Sem a sua motivação, força e perseverança este percurso tinha sido mais difícil.

Por último e não menos importantes, um grande obrigado à minha família por todo o apoio, motivação e força incondicional. Especialmente aos meus avós por toda a paciência, energia e carinho que tiveram comigo, aos meus pais por toda a educação, ensinamentos e amor que me deram ao longo da vida, sem eles este percurso académico nunca tinha sido possível.

Resumo

Este documento descreve os processos e ferramentas implementados para permitir a execução de várias instâncias do simulador de propagação de fogo, FARSITE, em ambiente *cloud*.

Este trabalho pretende acelerar a simulação de propagação de fogo, com diferentes variações dos dados de entrada do FARSITE, para melhorar a previsão da propagação de um incêndio.

Os resultados obtidos ajudam as várias entidades envolvidas no combate e prevenção de incêndios florestais.

Palavras-chave: Incêndios Florestais, FARSITE, ambientes *cloud*, simuladores de incêndios, modelos de incêndios.

Abstract

This work focuses on how we can implement a fire simulator on a *cloud*-based environment to assist all entities in combat and prevention of forest fires. Forest fires are quite destructives and random, due to these factors forest fires tend to cause economic and social matters.

In this document we present a solution based on a fire simulator that was implemented in the United States of America called FARSITE and we will be implementing it on a *cloud*-based environment.

With this work we want get new tools to the entities that will help to prevent and combat forest fires, In an easy and agile manner.

Keywords: Forest fires, FARSITE, *cloud*-based environment, fire simulator, fire-behaviour models.

INTRODUÇÃO	1
1.1 MOTIVAÇÃO	1
1.2 OBJECTIVO.....	1
1.3 ESTRUTURA DA DISSERTAÇÃO	2
ESTADO DE ARTE.....	5
2.1 MODELOS DE COMPORTAMENTO DE INCÊNDIO.....	5
2.1.1 <i>Modelo Rothermel 1972</i>	6
2.1.2 <i>Modelo Van Wagner 1977</i>	7
2.1.3 <i>Modelo Albin 1979</i>	8
2.1.4 <i>Modelo Rothermel 1983</i>	8
2.1.5 <i>Modelo Rothermel 1991</i>	9
2.1.6 <i>Modelo Nelson 2000</i>	9
2.1.7 <i>Modelo Scott e Reinhardt 2005</i>	10
2.2 TRABALHOS REALIZADOS EM SIMULADORES DE INCÊNDIO	11
2.2.1 <i>BehavePlus</i>	11
2.2.2 <i>FlamMap</i>	12
2.2.3 <i>FARSITE</i>	12
2.2.4 <i>Fire Characteristics Chart</i>	13
2.3 AMBIENTE CLOUD	13
2.3.1 <i>Benefícios da utilização Cloud</i>	14
2.3.2 <i>Modelos de Cloud</i>	15
2.3.3 <i>Serviços na cloud</i>	18
2.3.4 <i>Fornecedores cloud</i>	20
2.4 ANÁLISE AO ESTADO DE ARTE.....	21
IMPLEMENTAÇÃO	23
3.1 FARSITE	23
3.1.1 <i>Funcionamento</i>	24
3.1.2 <i>FARSITE no ambiente cloud</i>	25
3.2 INTERFACE GRÁFICA DO UTILIZADOR	27
3.2.1 <i>Web App e Interface gráfica</i>	27
RESULTADOS	33
CONCLUSÃO E TRABALHOS FUTUROS	42
5.1 CONCLUSÃO.....	42
5.2 TRABALHOS FUTUROS.....	44
BIBLIOGRAFIA.....	46
ANEXOS.....	48
7.1 INICIALIZAÇÃO INSTÂNCIA NA AWS	48

7.2	ACEDER À INSTÂNCIA VIA PUTTY	53
7.3	FARSITE.....	56

FIGURA 2.1.1 - DIAGRAMA DO MODELO DE ROTHERMEL 1972.....	7
FIGURA 2.1.2 - DIAGRAMA DO MODELO DE VAN WAGNER 1977	7
FIGURA 2.1.3 - DIAGRAMA DO MODELO ALBINI 1979.....	8
FIGURA 2.1.4 - DIAGRAMA DO MODELO DE ROTHERMEL 1983.....	9
FIGURA 2.1.5 - DIAGRAMA DO MODELO DE ROTHERMEL 1991.....	9
FIGURA 2.1.6 - DIAGRAMA DO MODELO DE NELSON 2000.....	10
FIGURA 2.1.7 - DIAGRAMA DO MODELO SCOTT REINHART 2005.....	11
FIGURA 2.3.1 - SERVIÇOS DE <i>CLOUD</i> E SERVIÇOS QUE PROVIDENCIAM. [13].....	20
FIGURA 3.1.1 - DIAGRAMA DA SIMULAÇÃO LOCALMENTE DO FARSITE.	25
FIGURA 3.1.2 - DIAGRAMA DA SIMULAÇÃO DO FARSITE NA INSTÂNCIA EC2.....	26
FIGURA 3.2.1 - CASE USE DA WEB APP.	28
FIGURA 3.2.2 - WEB APP SEPARADOR UM.	29
FIGURA 3.2.3 - WEB APP SEPARADOR DOIS.....	30
FIGURA 3.2.4 - WEB APP SEPARADOR TRÊS.	31
FIGURA 3.2.5 - WEB APP SEPARADOR QUATRO.	31
FIGURA 4.1 - WEB APP, COLOCAÇÃO DE UM TIPO DE FICHEIRO NA INSTÂNCIA.	34
FIGURA 4.2 - PUTTY, CONFIRMAÇÃO DA COLOCAÇÃO DOS FICHEIROS NA INSTÂNCIA.....	34
FIGURA 4.3 - PUTTY, FORMATO DO OUTPUT DA SIMULAÇÃO BEM SUCEDIDA NO SEGUNDO SEPARADOR DA WEB APP.....	36
FIGURA 4.4 - PUTTY, VISUALIZAÇÃO DO TEMPO DE RESPOSTA NO FICHEIRO DE SAÍDA TIMINGS.....	36
FIGURA 4.5 - PUTTY, SIMULAÇÃO EM PARALELO COM 3 LINHAS SEM VARIAÇÃO.	37
FIGURA 4.6 - À ESQUERDA VALORES ORIGINAIS DA VELOCIDADE DO VENTO E À DIREITA VALORES COM UMA VARIAÇÃO MÁXIMA DE 6%.....	38
FIGURA 4.7 - PUTTY, SIMULAÇÃO EM PARALELO COM 3 LINHAS ORIGINAIS COM VARIAÇÃO.....	39
FIGURA 7.1.1 - VISUALIZAÇÃO GERAL DE UMA INSTÂNCIA EC2.....	49
FIGURA 7.1.2 - EC2, PRIMEIRO PASSO ESCOLHA DO SISTEMA OPERATIVO.	50
FIGURA 7.1.3 - EC2, SEGUNDO PASSO ESCOLHA DO TIPO DE INSTÂNCIA.....	50
FIGURA 7.1.4 - EC2, TERCEIRO PASSO ESCOLHA DE CONFIGURAÇÕES AVANÇADAS.	51
FIGURA 7.1.5 - EC2, QUARTO PASSO ALOCAÇÃO DE MEMÓRIA.	51
FIGURA 7.1.6 - EC2, QUINTO PASSO CONFIGURAÇÃO DE SEGURANÇA.	52
FIGURA 7.1.7 - EC2 CONCLUSÃO DA CONFIGURAÇÃO DA INSTÂNCIA E CRIAÇÃO DO FICHEIRO PEM	53
FIGURA 7.2.1 - PUTTY COLOCAÇÃO DO FICHEIRO PPK.....	54
FIGURA 7.2.2 - PUTTY, CRIAR SESSÃO E COLOCAR DNS <i>PUBLICO</i> DA INSTÂNCIA.....	54
FIGURA 7.2.3 - PUTTY, LIGAÇÃO AO TERMINAL DA INSTÂNCIA VI PUTTY.....	55
FIGURA 7.3.1 - FORMATO DO FICHEIRO RUNPANTHER.	56
FIGURA 7.3.2 - LINHA DE COMANDO PARA CORRER O SIMULADOR.	56

Lista de Tabelas

TABELA 2.1.1- MODELOS DE PROPAGAÇÃO, COMPORTAMENTO E DETECÇÃO DO FOGO.	6
TABELA 2.3.1 - BENEFÍCIOS DE USO DE UMA <i>CLOUD</i>	14
TABELA 2.3.1-MODELOS <i>CLOUD</i> E SUAS DESCRIÇÕES E EXEMPLOS.	15
TABELA 2.3.2- VANTAGENS E DESVANTAGENS DOS MODELOS <i>CLOUD</i>	16
TABELA 2.3.1- SERVIÇO DE <i>CLOUD</i> E BREVE DESCRIÇÃO.	19
TABELA 2.3.1- FORNECEDORES E TIPOS DE <i>CLOUD</i> QUE OFERECEM.	21
TABELA 7.1.1 - EC2 NOMES DE UTILIZADORES PADRÃO PARA ACESSO DE SSH.....	52

Lista de acrónimos

AWS – Amazon Web Service.

CSP - Cloud Service Provider.

GIS – Sistema de Informação Geográfica. [Geographic Information System]

IaaS – Estrutura como serviço. [Infrastructure as a Service]

IBM - international Business Machines.

LCP - Ficheiro de Terreno.

PaaS – Plataforma como serviço. [Platform as a Service]

Pay-as-usage-fee – Pagamento por uso.

SaaS – Software como serviço [Software as a Service]

SHP – Ficheiro Shapefile.

SHX – Ficheiro Shapefile Index.

SSH – Secure Shell.

VNC – Virtual Network Computing.

Web App – Aplicação Web.

1

Introdução

1.1 Motivação

Os incêndios são uma grande preocupação pois alteram ecossistemas e têm implicações económicas e sociais, destroem habitações, campos agrícolas, fábricas, floresta e flora. Existem vários tipos de incêndios, mas abordaremos os incêndios florestais que são os mais destrutivos, incontrolláveis e aleatórios.

Os incêndios florestais têm vários factores que determinam e influenciam o seu comportamento e propagação: temos os comburentes (e.g árvores e vegetação), combustíveis (e.g cigarros, queimadas, natural ou não natural), território e o seu ordenamento e condições climáticas (e.g humidade, temperatura, vento).

1.2 Objectivo

Pretende-se provar que um simulador de incêndios seja capaz de trabalhar em ambiente *cloud* e que seja capaz de executar várias simulações em simultâneo de maneira ser usado pelas devidas entidades no combate e prevenção a incêndios florestais (e.g bombeiros, protecção civil).

Provando este conceito, o simulador a funcionar em ambiente *cloud*, todas as entidades que combatem os incêndios florestais poderão usufruir deste simulador para poderem atuar em qualquer parte de Portugal e assim terem mais uma

ferramenta na ajuda para preverem o comportamento e propagação de um incêndio.

1.3 Estrutura da dissertação

Esta secção serve de apresentação ao trabalho que irá ser realizado na dissertação, como está dividido e qual a sua estrutura. O documento é composto por quatro capítulos, estado de arte, implementação, resultados e por último conclusão e trabalhos futuros respectivamente.

O capítulo dois está organizado de maneira a contextualizar todos os temas abordados nesse mesmo capítulo. Na primeira parte são introduzidos os modelos de comportamento e propagação de um incêndio florestal, referindo o autor do modelo, a data de *publicação*, entradas do modelo e suas saídas e um diagrama sobre o funcionamento do modelo. Na segunda parte aborda-se os simuladores de incêndios, apresentando assim o sistema operativo de cada simulador, modelos utilizados, funcionamento e as suas respectivas entradas e saídas. Já na terceira parte deste capítulo é abordado os ambientes *cloud*, onde se comenta sobre os benefícios, modelos, serviços e fornecedores dos ambientes *cloud*, por fim temos a análise do estado de arte onde indicamos as escolhas que serão tomadas para a implementação.

A implementação é feita no capítulo três onde é falado de como iremos abordar o tema desta dissertação. É neste capítulo que se fala com mais detalhe o funcionamento do simulador de incêndio localmente e em ambiente *cloud* e como o simulador foi integrado nesse mesmo ambiente. Abordamos também como é possível tornar o uso do ambiente *cloud* com o simulador mais simples e rápido para o utilizador final. O capítulo quatro consiste em vários testes à implementação realizada anteriormente e se verifica, se esta se encontra com algum problema ou se correu tudo dentro do implementado. Por último, no capítulo cinco, temos a conclusão e trabalhos realizados, é neste capítulo que se descreve como a implementação e os resultados correram e quais os problemas e desafios

encontrados ao longo da implementação, no fim fala-se de trabalhos que poderão ser realizados no futuro como melhoramento a esta implementação.



Estado de Arte

O capítulo dois é composto por toda informação e pesquisa relacionada com o tema do documento. É a partir deste capítulo que são realizadas as escolhas necessárias para o desenvolvimento da implementação. Este capítulo está dividido em quatro partes sendo elas modelos de comportamentos de incêndios, simuladores de incêndio, ambientes *cloud* e análise ao estado de arte.

A primeira parte deste capítulo abordamos os modelos de incêndio, quem os criou e uma breve descrição e esquema do modelo. Na segunda parte falasse dos simuladores de incêndios usados para o estudo do comportamento do fogo e quais os modelos que usam para as suas simulações. Na terceira parte estão descritos os serviços e tipos de ambientes *cloud*, por último temos a análise ao estado de arte onde é mencionado o que se vai utilizar deste capítulo na implementação.

2.1 Modelos de Comportamento de Incêndio

Para melhor se saber com algum rigor como o fogo se propaga e comporta, num determinado local e tempo, é necessário ter em conta os seguintes aspectos: como este é formado, em que tipo de terreno nos encontramos, tipo de vegetação ou flora que temos no terreno e quais as condições climáticas. Foram criados modelos matemáticos com o propósito de simplificar o estudo do comportamento e propagação de um incêndio. Um modelo matemático tem como objectivo a descrição,

análise e representação de um sistema complexo em um sistema mais simples, fácil de entender e de replicar.

Pesquisou se modelos matemáticos de comportamento e propagação de incêndio com base nesta pesquisa encontraram-se os modelos referidos na tabela 2.1.1 em baixo e concluiu-se que nem todos os modelos tem o mesmo propósito, a maioria dos modelos descreve a propagação de um incêndio em vários tipos de flora. Na seguinte tabela 2.1.1 encontram-se todos os modelos observados neste estudo e uma breve descrição e funcionalidade.

Tabela 2.1.1- Modelos de propagação, comportamento e detecção do fogo.

Modelo	Descrição
Rothermel (1972)	Propagação de um fogo de superfície. [1]
Van Wagner (1977)	Inicialização de um fogo de copa. [2]
Albini (1979)	Detecção de um fogo. [3]
Rothermel (1983)	Propagação e intensidade de um fogo. [4]
Rothermel (1991)	Propagação de um fogo de copa. [5]
Nelson (2000)	Humidade em comburentes mortos.
Scott e Reinhardt (2005)	Avaliação do fogo de copa.

2.1.1 Modelo Rothermel 1972

O modelo de Rothermel, de 1972, é um modelo matemático usado para prever a propagação e intensidade de um fogo de superfície. Com este modelo não é possível prever se o incêndio chega às copas das árvores, porque este modelo só tem em consideração os comburentes que se encontrem à superfície.

Este modelo tem como entradas a composição química e física dos comburentes e as condições climáticas quando ocorre a ignição, também podemos ter como entradas adicionais face ao comburente os seguintes dados: peso, rácio entre área e

volume, calor por partícula, humidade, conteúdo de minerais e também a humidade à qual se espera extinção do fogo. Face às entradas climáticas e territoriais temos de ter em conta a velocidade do vento e inclinação do terreno respectivamente. [1]

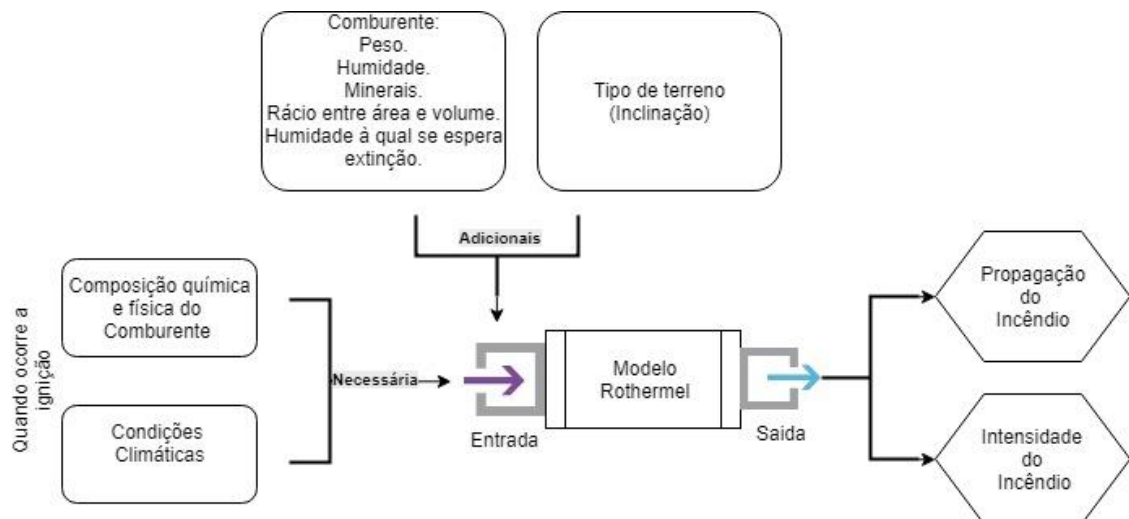


Figura 2.1.1 - Diagrama do Modelo de Rothermel 1972

2.1.2 Modelo Van Wagner 1977

Van Wagner em 1977, criou um modelo matemático sobre como um fogo de copa se pode iniciar e propagar. É denominado fogo de copa ao fogo que atinge e arde nas copas das árvores. Este modelo toma como princípio o volume e humidade que existe nas folhas das árvores e separa os comburentes que existem à superfície dos comburentes das copas das árvores, tratando do último comburente como um único. [2]

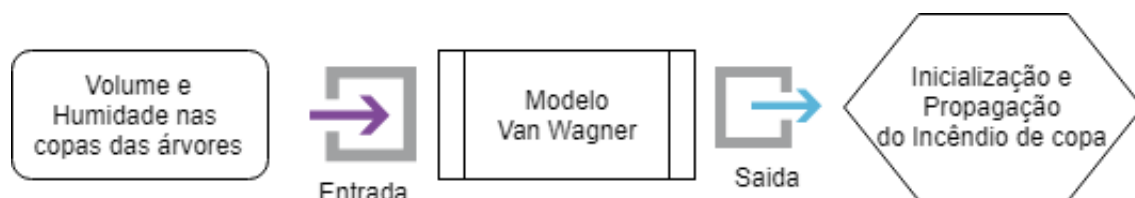


Figura 2.1.2 - Diagrama do Modelo de Van Wagner 1977

2.1.3 Modelo Albini 1979

O modelo matemático de Albini foi criado em 1979 e diz-nos a que distância máxima podemos detectar um incêndio consoante o seu tipo. Este modelo usa como entrada três tipos de fontes de incêndios: incêndios de copas ou superfície e pilhas/queimadas. O modelo usa o espectro electromagnético para distinguir cada um dos tipos de incêndios através da energia que emitem e assim calcular a sua distância máxima de visibilidade.

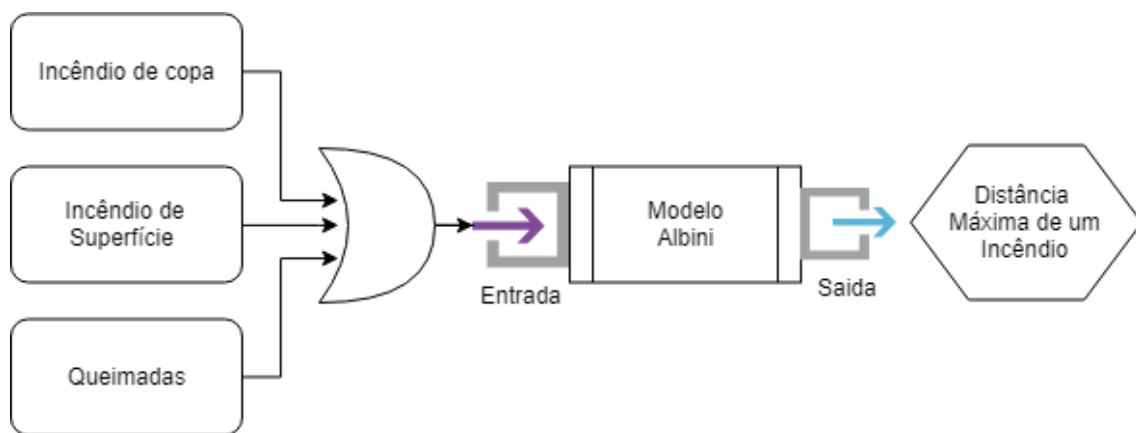


Figura 2.1.3 - Diagrama do Modelo Albini 1979

2.1.4 Modelo Rothermel 1983

O seguinte modelo criado por Rothermel em 1983 é um melhoramento ao já realizado em 1972, neste modelo conseguimos, para além de prevermos a propagação e comportamento, também sabe a intensidade de um incêndio. Este modelo tem como entradas: tipo de comburente, humidade do comburente morto e vivo, inclinação do terreno e direcção e velocidade do vento, nas suas saídas temos: intensidade e propagação do incendio e calor gerado por unidade de área.

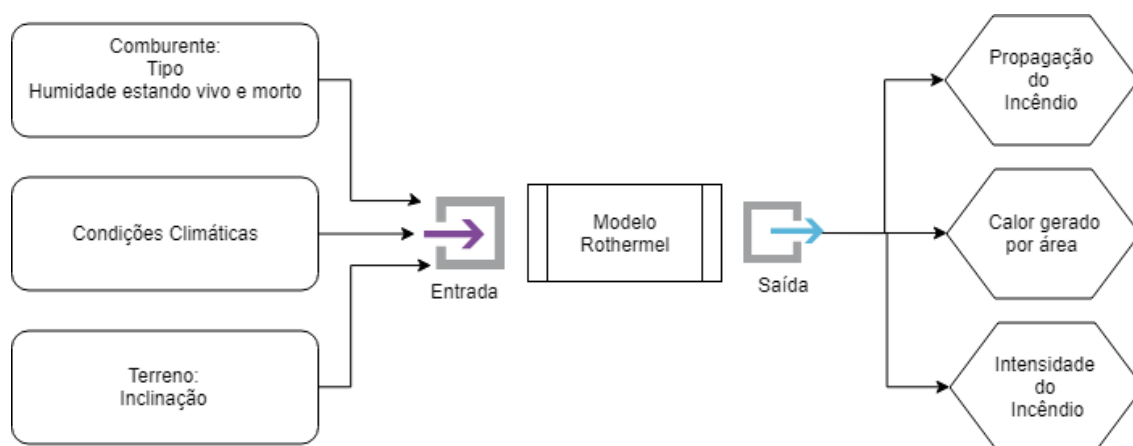


Figura 2.1.4 - Diagrama do Modelo de Rothermel 1983

2.1.5 Modelo Rothermel 1991

Rothermel realizou outro modelo em 1991 em relação ao incêndio de copa. Este modelo prevê a propagação, intensidade e tamanho de um incêndio de copa de maneira a aproximar o comportamento do mesmo consoante os seus comburentes, condições climáticas e tipo de terreno.

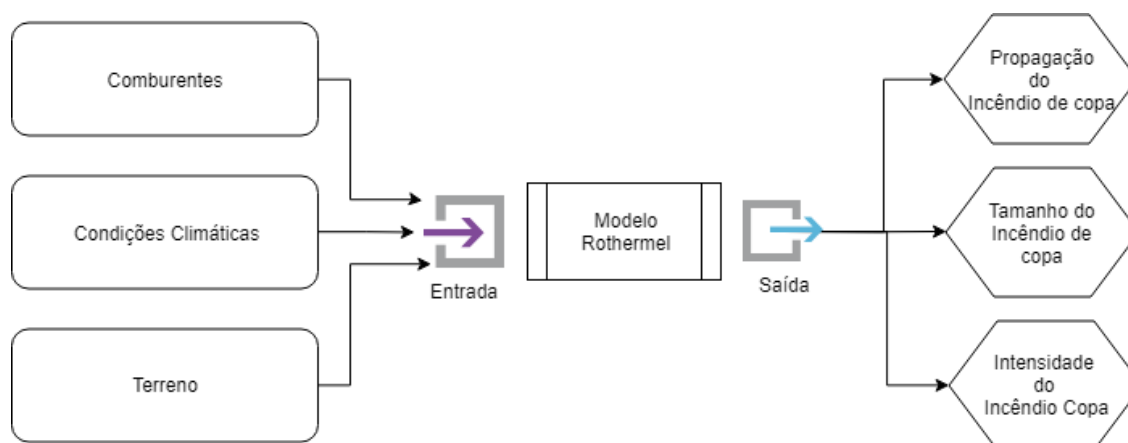


Figura 2.1.5 - Diagrama do Modelo de Rothermel 1991

2.1.6 Modelo Nelson 2000

O modelo de Nelson foi criado em 2000 com o intuito de classificar a humidade existente em comburentes mortos. É um modelo físico que contém as equações de

calor, humidade e dados referentes à água no comburente e solo. Como entradas tem: temperatura do ar, humidade relativa, irradiância e a última entrada da precipitação do local, na sua saída temos: níveis de humidade e temperatura dos combustíveis.

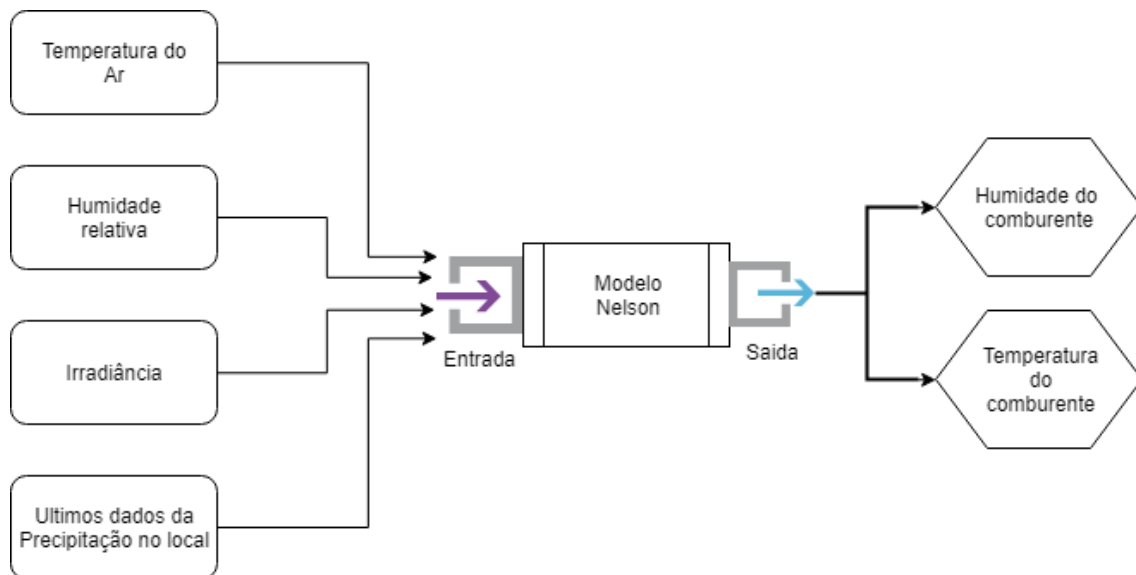


Figura 2.1.6 - Diagrama do Modelo de Nelson 2000

2.1.7 Modelo Scott e Reinhardt 2005

Por último temos o modelo de Scott Reinhardt criado em 2005, o modelo avalia a suscetibilidade de existir fogo de copa num incêndio e também é capaz de estimar o comportamento do incêndio, mas não foi com este propósito que ele foi criado.

O modelo usa como entradas os modelos de Rothermel (1972 e 1991) e também o modelo de Van Wagner (1977) pois estes modelos são os usados com mais frequência no comportamento e propagação de um incêndio.

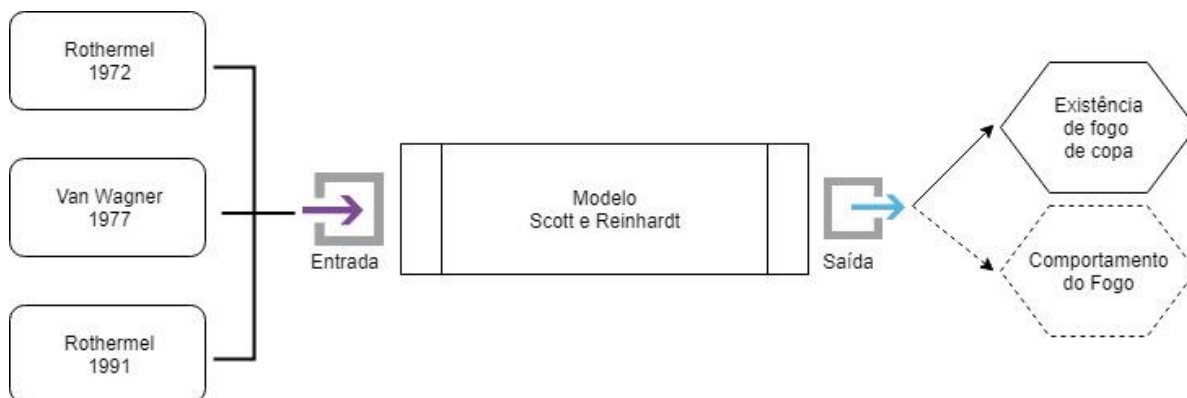


Figura 2.1.7 - Diagrama do Modelo Scott Reinhardt 2005

2.2 Trabalhos Realizados em Simuladores de Incêndio

Nesta secção iremos falar com mais algum detalhe do que realmente irá ser tratado neste documento. Iremos abordar simuladores de incêndio florestal, como funcionam, as suas vantagens e desvantagens e quais os algoritmos que são usados.

Um incêndio florestal pode ser definido de inúmeras maneiras, nesta secção iremos ter em foco o comportamento e sua propagação do fogo durante um incêndio, pesquisaram se programas capazes de definir e prever o comportamento do fogo num determinado local, encontrámos quatro programas com esta capacidade e que estão descritos em baixo.

2.2.1 BehavePlus

BehavePlus é um programa que toma como base o sistema operativo Windows e que pode ser usado para calcular o comportamento do fogo por qualquer sistema de controlo de incêndios.

Usa o comburente e a humidade na simulação e classificação de incêndios de superfície e copa, intensidade e progressão do fogo, local da ignição, tamanho, taxa de mortalidade das árvores e distância de visão. Este programa só possibilita a recolha desta informação pois usa como algoritmo o método de Scott e Reinhardt (2005). [6]

2.2.2 FlamMap

FlamMap é desenvolvido para todos os sistemas operativos e descreve uma possibilidade do comportamento do fogo tendo em conta as seguintes entradas: condições climáticas, comburentes e humidade.

Este programa usa uma imagem satélite (GIS) do local para calcular em cada pixel de forma independente o comportamento do fogo, não podendo assim calcular a propagação do fogo. Podemos contar com os modelos de Rothermel (1972 e 1991), Van Wagner (1977) e Nelson (2000) neste programa. O FlamMap não substitui o FARSITE pois não inclui nas suas simulações as variações temporais do fogo que são causadas pelas condições climáticas e flutuações diurnas.

A grande desvantagem deste programa é que não actualiza as suas entradas, ou seja, depois de inseridas estas são tratadas e o programa diz uma possibilidade do comportamento do fogo. [7]

2.2.3 FARSITE

FARSITE também é desenvolvido para trabalhar em todos os sistemas operativos e simula o crescimento e comportamento de um incêndio durante determinados períodos de tempo.

Para as suas simulações o FARSITE precisa de uma imagem topográfica, comburentes, condições climáticas e vento do local durante a ocorrência. Usa como modelos de comportamento do incêndio, Rothermel (1972 e 1991), Van Wagner (1977), Albini (1979) e Nelson (2000).

Este programa é um modelo determinista, podendo assim usar a sua simulação para comparar com as entradas e se poder actuar conforme as conclusões retiradas e também podemos colocar nas simulações outros comportamentos de maneira a compreender o fogo nesses mesmos comportamentos, mesmo não sendo o que se esta a passar no local naquele momento. [8]

2.2.4 Fire Characteristics Chart

Fire Characteristics Chart faz parte do BehavePlus, mas aqui é apresentado como um programa independente. É utilizado para representar graficamente o comportamento e o risco do incêndio de maneira a se poder apresentar estes mesmos gráficos de forma mais clara ao público com menos conhecimento em relação a incêndios.

Com este programa podemos ver um incêndio modelado ou observado de acordo com tamanho da sua chama, zona ardida por área bem como velocidade de propagação, com estas entradas o programa poderá assim desenhar os gráficos do comportamento de um incêndio de superfície ou copa ajudando assim nos briefings que ocorrem no planeamento do combate a um incêndio.

2.3 Ambiente *cloud*

A *cloud* proporciona serviços computacionais em qualquer parte do mundo desde que o utilizador ou empresa tenha acesso à internet. A *cloud* pode servir para várias funções tais como servidores, centros de dados, capacidade de armazenamento, software entre outros. Assim sendo a empresa ou utilizador não necessitará que o serviço esteja no disco rígido do seu computador ou que esteja ligado à mesma rede interna, só necessitando de uma ligação à internet para usar o serviço requisitado. Conseguindo assim que o mesmo tipo de serviço oferecido por uma *cloud* seja utilizado por milhões de empresas ou utilizadores diferentes. [9], [10]

Definição de *cloud*

"A Cloud is a type of parallel and distributed system consisting of a collection of inter-connected and virtualized computers that are dynamically provisioned and presented as one or more unified computing resource(s) based on service-level agreements established through negotiation between the service provider and consumers."
[9]

2.3.1 Benefícios da utilização *Cloud*

As *cloud* tem custos reduzidos para as empresas pelo simples facto de não precisarem de um local físico nas empresas e de não obrigarem as mesmas a terem hardware e software capaz de armazenar toda a informação, executar programas ou aplicações que produzam guardada nos seus computadores ou em data centers. Por não precisarem de um local físico na empresa e de não obrigar à empresa um constante melhoramento de hardware e software as *Clouds* tem como benefício a produtividade pois todo o pessoal da empresa poderá ser produtivo deixando este tipo de trabalho para o fornecedor. Uma *cloud* apresenta uma velocidade elevada pois providência os seus serviços conforme a demanda pela empresa o que não é tão exigente para o CPU ou memória porque só irá usar o que for requisitado pelo utilizador.

Com uma *cloud* a empresa poderá criar ou deslocar-se para qualquer parte do mundo desde que exista uma conexão à internet conseguindo assim uma enorme vantagem para um rápido desenvolvimento da empresa caso exista. Nos últimos dois benefícios das *Clouds*: desempenho e segurança, ficam ao cargo dos fornecedores que se asseguram através de melhoramentos ou criação de novos de equipamentos (hardware e software) e data centers para o desempenho das suas *Clouds* bem como criando políticas, tecnologias e controlos para as empresas para estas terem maior segurança. Na tabela em baixo (Tabela 2.3.1) temos os benefícios das *Clouds* e também uma breve descrição dos mesmos. [11], [12]

Tabela 2.3.1 - Benefícios de uso de uma *Cloud*.

Custos	Baixa manutenção de hardware ou software. [11], [12]
Velocidade	Não requer espaço físico no computador ou data centers. [11], [12]
Crescimento global	Rápido acesso em qualquer parte do mundo. [11], [12]

Produtividade	Empregados focam-se na produção e não na manutenção. [11], [12]
Desempenho	Melhor hardware e software do mercado providenciado pelo fornecedor. [11], [12]
Segurança	Políticas, controlo e tecnologias oferecidas pelo fornecedor. [11], [12]

2.3.2 Modelos de *Cloud*

Existem quatro modelos de *cloud* que os fornecedores fornecem a qualquer utilizador e empresa. Na tabela 2.3.2 encontra-se uma breve descrição de cada modelo de *cloud* bem como exemplos e comerciais das mesmas.

Tabela 2.3.2-Modelos *Cloud* e suas descrições e exemplos.

Modelo	Descrição	Exemplo
<i>Public Cloud</i>	Domínio <i>publico</i> . Baixos custos e manutenção. [10]–[13]	Amazon Elastic Compute <i>Cloud</i> (EC2), IBM's Blue <i>Cloud</i> , Sun <i>Cloud</i> , Google AppEngine and Windows Azure Services Platform. [14]–[17]
<i>Private Cloud</i>	Domínio privado. Maior segurança, privacidade. Pode ser colocada ou não num local físico. [10]–[13]	Qualquer empresa/Companhia para uso interno. Eucalyptus, Ubuntu Enterprise <i>Cloud</i> , Amazon VPC (Virtual <i>Private Cloud</i>), VMware <i>Cloud</i> , Infrastructure Suite, Microsoft ECI data center. [14]–[17]
<i>Hybrid Cloud</i>	Composto pelo menos por uma <i>Public Cloud</i> e uma <i>Private Cloud</i> .	Adobe Creative <i>Cloud</i> , Sage200 online, Windows Azure, VMware <i>Cloud</i> .

	Utilizadores são tratados como entidades diferentes uns dos outros. [10]–[13]	[14], [16], [17]
<i>Community Cloud</i>	Ainda em desenvolvimento. É um meio termo entre uma <i>Public Cloud</i> e <i>Private Cloud</i> . Usado por empresas/companhias que partilhem a mesma privacidade, segurança e regulamentação. [10]–[12], [18]	Google Apps for Government, Microsoft Government <i>Community Cloud</i> , e empresas estatais como a Saúde. [14], [16], [17]

Na tabela 2.3.3 apresentamos as vantagens e desvantagens dos modelos referidos em cima.

Tabela 2.3.3- Vantagens e desvantagens dos modelos *cloud*.

Modelo	Vantagens	Desvantagens
<i>Public Cloud</i>	Custo. Manutenção. Fácil montagem. [12], [15]	Segurança. Controlo. [12], [15]
<i>Private Cloud</i>	Segurança. Privacidade. Controlo. [12], [15]	Custo. Manutenção. [12], [15]

<i>Hybrid Cloud</i>	Controlo. Alocação de recursos para novos projetos. Optimização da infraestrutura. Custos. [12], [15]	Segurança. Privacidade. [12], [15]
<i>Community Cloud</i>	Custo. Manutenção. [12], [15]	Segurança. Controlo. Recursos partilhados com todos os membros. [12], [15]

As *Public Clouds* são de domínio público, de baixos custos e de manutenção para as empresas ou utilizadores isto acontece pelo facto de estas *clouds* não se encontrarem num local físico. Poderão representar um enorme crescimento empresarial pois cada utilizador que usar a *cloud* para um determinado serviço poderá, ou não, ser cobrado pelo uso exclusivo do mesmo, pagando pelo uso (pay-per-usage fee). Os fornecedores deste modelo fazem manutenção e gestão das infraestruturas e recursos usados pelos utilizadores ou empresas. Uma grande vantagem deste modelo é a da sua rápida montagem e acessibilidade. As suas desvantagens serão essencialmente de segurança e controlo, devido ao seu domínio ser público e ser um alvo fácil de ataques pois torna-se difícil controlar todos utilizadores que usam a *cloud* pois é o fornecedor que faz grande parte do trabalho referente a este tópico. [10] –[13], [15]

A *Private Cloud*, como o nome indica, apresenta um domínio privado onde nenhum utilizador ou empresa poderá aceder, a não ser que faça parte do mesmo domínio. Este modelo de *cloud* pode ter um local físico, ou não, e pode ser controlado e gerido pela empresa ou por uma outra empresa que seja contratada para esse efeito. Uma das grandes vantagens deste modelo é a sua segurança, privacidade e controlo da *cloud*, por outro lado apresenta maiores custos de manutenção, especialmente se tiver um local físico para a empresa. [10]–[13], [15]

Em terceiro lugar temos as *Hybrid Cloud* que são uma composição de duas infraestruturas *cloud* diferentes entre as *Private*, *Public*, *Community*. Normalmente uma *Hybrid Cloud* é composta por uma *Private Cloud* e uma *Public Cloud*. Neste modelo todos os utilizadores são tratados como entidades diferentes uns dos outros, pois à medida que a demanda de recursos aumenta as empresas poderão passar esses recursos de domínio privado para domínio *publico* mitigando o seu trabalho. As *Hybrid Cloud* têm como vantagens, maior controlo e optimização sobre a infraestrutura da *cloud*, custos de manutenção e facilidade de alocação de recursos para novos projectos. As grandes desvantagens serão a privacidade e controlo quando os seus recursos estão em domínio *publico* devido ao facto de estar mais sujeitos a ataques. [10]–[13], [15]

Por último temos as *Community Cloud* que são um modelo ainda em desenvolvimento, mas pode se dizer que são um meio termo entre a *Public* e *Private Cloud*. Enquanto os outros modelos podem ser utilizados só por um utilizador ou empresa este modelo pode ser usado, gerido e controlado por várias empresas desde que estas partilhem as mesmas políticas de privacidade, segurança e regulamentação. Apresentam como vantagem o baixo custo e manutenção, face aos outros três modelos e como desvantagem temos segurança e controlo devido ao facto de este modelo ainda estar em desenvolvimento e todos os recursos da *cloud* terem de ser partilhados por todos em comunidade. [10] –[12], [18]

2.3.3 Serviços na *cloud*

As *cloud* foram divididas em serviços para qualquer utilizador ou empresa poder escolher a que melhor se adequa para os seus objectivos. A tabela 2.3.4 em baixo temos os serviços de *Clouds* existentes no mercado.

Tabela 2.3.4- Serviço de *Cloud* e breve descrição.

Serviço de <i>Cloud</i>	Descrição
<i>IaaS- Infrastructure as a Service</i>	Serviço de <i>Cloud</i> mais básico de todos. Serviços fundamentais. <i>Pay-as-usage-fee</i> . [10]–[12], [19], [20]
<i>PaaS- Platform as a Service</i>	Ferramentas e meios para criar os próprios programas ou aplicações. Comprar e correr aplicações ou programas criados por outros. [10]–[12], [19], [20]
<i>SaaS- Software as a Service</i>	Utilizadores tem a capacidade de poderem usar programas ou aplicações dos fornecedores. Não obriga a ter licenças de comercialização de determinado produto. [10]–[12], [19], [20]

O *IaaS* é o pilar de toda arquitectura e tecnologia *cloud* e é também com base neste serviço que os outros foram criados. Este serviço de *cloud* fornece os serviços fundamentais de uma *cloud*, armazenamento, servidores, rede a todos os consumidores finais.

Este serviço de *cloud* não é muito usado por empresas desenvolvidas, mas em contrário é usado por empresas (e.g startups e PME) que estejam a desenvolver-se assim as empresas poderão só pagar pelos serviços que usam (*Pay-as-usage-fee*) dando uma boa margem e flexibilidade económica às empresas. O pouco controlo que têm neste tipo de *cloud* é ao nível de segurança básica, poderão ter mais dependendo das aplicações ou programas que terão a correr na *cloud* tudo o resto referente à infraestrutura da *cloud* é tratada pelo fornecedor. [10]–[12], [19], [20]

O *PaaS* é o nível acima do serviço anterior de *cloud*, para além dos serviços essenciais que os fornecedores providenciam estes também oferecem um conjunto de ferramentas e meios (máquinas virtuais e sistemas operativos) para que os utilizadores possam criar os seus programas ou aplicações sendo estes obrigados às linguagens, bibliotecas, ferramentas e serviços suportados pelo fornecedor da *cloud*.

Os custos deste serviço de *cloud* são mais elevados que o *IaaS* devido a existência de máquinas virtuais que necessitam de ser protegidas contra os ataques, em termos de segurança e manutenção permanece tudo idêntico ao *IaaS*. [10] – [12], [19], [20]

Por último temos o *SaaS* é o último nível dos serviços de *cloud*, este serviço de *cloud* oferece o que os dois últimos oferecem com a adição de poderem correr aplicações ou programas fornecidos pelo fornecedor.

Este serviço de *cloud* tem vindo a crescer devido ao facto de os utilizadores poderem utilizar a internet de maneira a comercializarem os seus programas ou aplicações, assim o comerciante não precisa de ter licenças para efectuar a venda do seu produto, no entanto os fornecedores tem de garantir uma segurança elevada. [10]– [12], [19], [20]

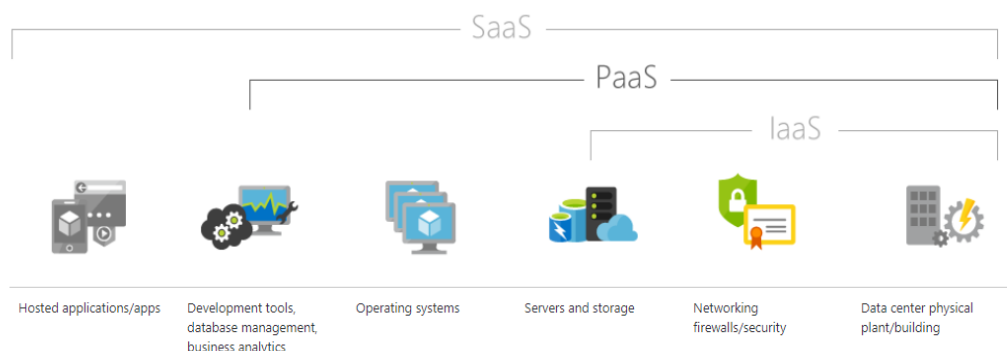


Figura 2.3.1 - Serviços de *cloud* e serviços que providenciam. [13]

2.3.4 Fornecedores *cloud*

Os fornecedores considerados neste estudo, são grandes empresas que tem já um grande impacto ou nome na internet, tais como a Google, Microsoft, Amazon, IBM ou Oracle entre outros. São estes fornecedores que têm vários data centers por todo o mundo e que oferecem a todos os que usufruírem dos seus serviços *cloud* redundância, segurança, fiabilidade e baixos custos de manutenção. Estes também oferecem inúmeros outros serviços que podem ser associados ao serviço *cloud*. Na tabela em baixo (Tabela 2.3.5) esta colocada alguns fornecedores e todos os serviços de *Clouds* que fornecem. [9], [21]

Tabela 2.3.5- Fornecedores e tipos de *Cloud* que oferecem.

<i>Cloud</i> Service Provider (CSP)	<i>IaaS</i>	<i>PaaS</i>	<i>SaaS</i>
Amazon	X	X	
Century Link	X	X	
Google	X	X	X
Oracle	X	X	X
IBM	X	X	X
Microsoft	X	X	X
Rackspace	X	X	
Salesforce.com		X	
SAP	X	X	X
Verizon Terremark	X	X	

2.4 Análise ao estado de arte

Com o estado de arte concluído e sabendo o que temos ao nosso dispor, iremos seguidamente escolher e justificar o que irá ser usado na nossa implementação do simulador de incêndios em ambiente *cloud*.

Em relação ao simulador de incêndios iremos usar o FARSITE pois é um programa completo e versátil, devido a usar a maioria dos modelos referenciados na secção 2.1, por ser um programa determinista deixando comparar as entradas com as saídas das suas simulações, podendo assim dar uma resposta rápida ao incêndio e também ajudando na prevenção de incêndios, visto que se pode usar o FARSITE para simular incêndios e ver que comportamento este toma. O FARSITE é um programa versátil pois pode trabalhar em todos os sistemas operativos sem qualquer tipo de restrição.

Iremos optar por uma *Public Cloud* devido aos seus baixos custos, rápida montagem e escalabilidade. Com este modelo de *cloud* conseguiremos chegar a qualquer entidade que poderá usar o programa de maneira a intervir e combater o incêndio com eficiência e rapidez. Caso existam dados que não poderão ser de domínio público teremos de passar para uma *Hybrid Cloud* de maneira a proteger esses dados, com a contrapartida de este modelo ter um custo mais elevado. Qualquer que seja o modelo utilizado de *cloud* precisaremos sempre de usar um *PaaS (Platform as a Service)*, pois é este tipo de *cloud* que nos dá liberdade para criar e inserir um programa num serviço *cloud* e poder ter acesso a máquinas virtuais.

O fornecedor escolhido para a implementação foi a Amazon (AWS, Amazon Web Services), pelo facto de fornecer 12 meses gratuitos após a inscrição. Nesses 12 meses o utilizador pode usar todos os serviços gratuitos que a AWS fornece desde que não ultrapasse os recursos alocados para os níveis gratuitos, após exceder esse limite pagará taxas conforme o uso. O nível gratuito da AWS é bastante razoável pois oferece todos os serviços que a própria AWS fornece a todos os utilizadores não gratuitos, mas com certas restrições dando assim a todos os utilizadores gratuitos a possibilidade de usufruírem de todos os serviços da AWS.

3

Implementação

Neste capítulo iremos abordar como o simulador funciona localmente, na instância, e como os utilizadores finais poderão utilizar o simulador na instância através de uma web app como interface gráfica de utilizador. Na primeira secção descreveremos com rigor quais as entradas que o FARSITE utiliza e como é que podemos utilizar essas mesmas entradas para realizar uma simulação seja ela localmente ou na instância. Na segunda secção descrevemos com que recursos a web app foi criada, como os utilizadores finais poderão ligar-se e como a web app funciona e quais as novas funções implementadas que o FARSITE não seria capaz de fazer sem recurso a esta web app.

3.1 FARSITE

O FARSITE é um programa de simulação de incêndios capaz de simular o comportamento e propagação de um incêndio. Este programa recorre a três tipos de ficheiros de entrada shp, input, lcp que depois da simulação geram um conjunto de ficheiros output consoante as entradas colocadas no ficheiro RunPanther.

O ficheiro *Shapefile* (shp) é um tipo ficheiro que contém dados espaciais em formato vectorial de uma área, e este tem sempre outro tipo de ficheiro associado, o

Shapefile index (shx) no qual classifica todos os tipos de linhas e formas que o ficheiro principal shp contem.

O tipo de ficheiro input é onde é colocada toda a informação relacionada com o início do incêndio (ignições, duração), níveis de humidade do local e da respectiva flora, tipo de folhagem e árvores, meteorologia do local e também quais os modelos de incêndio das copas.

Por último o ficheiro *landscape* (lcp) que contém a informação referente à localização (latitude), topologia (inclinação, elevação) e todas as unidades necessárias para as medições efectuadas no simulador.

3.1.1 Funcionamento

Através da linha de comandos temos de ir à pasta *src* do FARSITE e realizar um *make* dessa mesma pasta, devemos criar um executável para realizarmos uma simulação no FARSITE. Antes de se efectuar uma simulação é necessário criar um ficheiro de texto (runPanther.txt) que indique as directorias onde estão os três ficheiros de entrada e onde se indica também a directoria onde queremos guardar as saídas. Devemos separar cada tipo de informação (entrada ou saída) colocando '0' após o nome do ficheiro de maneira a que o programa saiba onde acabam as entradas e começam as saídas até ao final do documento. Consultar anexo 7.3 figura 7.3.1.

Com o ficheiro de texto criado é preciso aceder à linha de comandos e executar uma instrução, essa instrução tem de indicar onde está a directoria do executável bem como directoria do ficheiro de texto criado que contem os três ficheiros de entrada e a pasta de saída, consultar anexo 7.3 figura 7.3.2.

Na figura 3.1.1 temos um diagrama de como é que o FARSITE corre localmente com os ficheiros de entrada pretendidos devidamente escritos no ficheiro de texto e saídas da simulação.

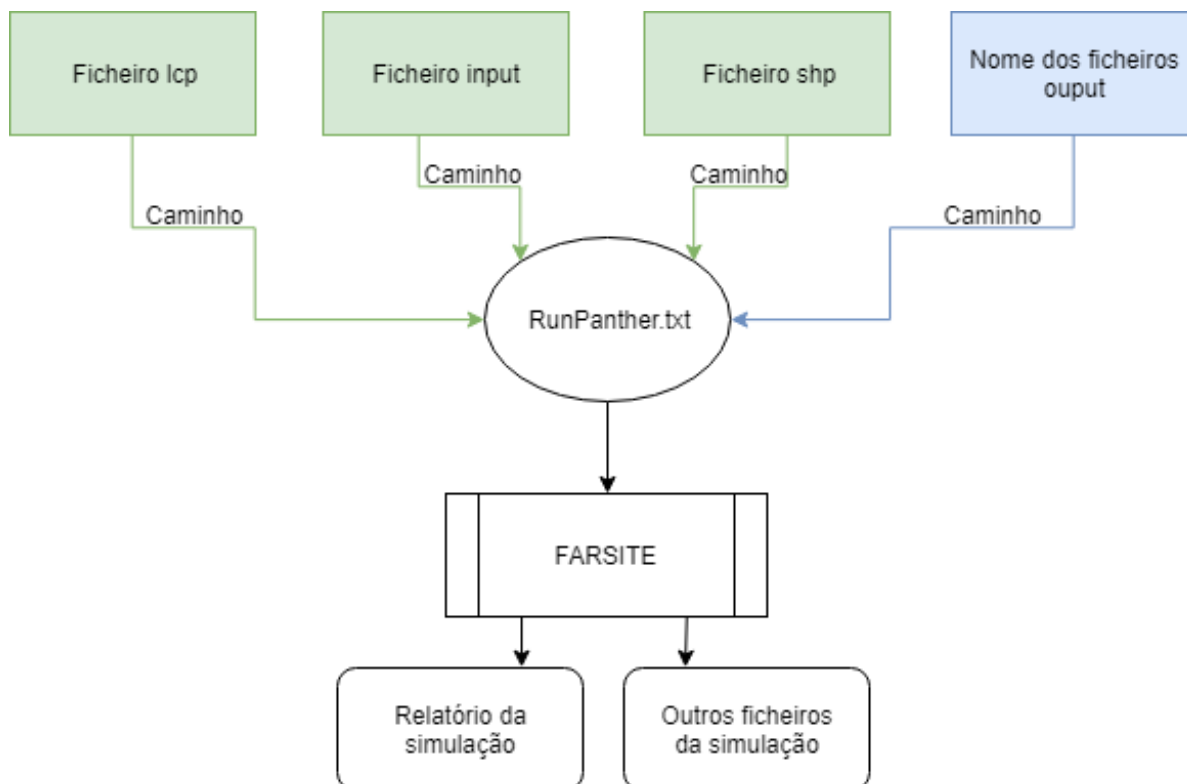


Figura 3.1.1 - Diagrama da simulação localmente do FARSITE.

3.1.2 FARSITE no ambiente *cloud*

O FARSITE foi copiado para a instância através do comando SCP (secure copy), este comando passa um único ficheiro para uma máquina remota. Para fazer uma cópia para uma máquina remota utiliza-se o seguinte comando: *scp FicheiroEnviar NomeUtilizador@DNS_Publico:DirectoriaDaCopia*.

Neste caso como iremos copiar ficheiros para uma instância da AWS o scp tem de ser alterado pois a chave encriptada da AWS tem de ser inserida na linha de comando senão não irá copiar os ficheiros, ficando então o seguinte comando a ser utilizado: *scp -i ChaveEncriptada FicheiroEnviar ubuntu@DNS_Publico:/home/ubuntu*. Como o scp não lida com o envio de pastas foi necessário criar um zip do FARSITE para evitar repetição do scp e/ou enviar ficheiros importantes do simulador para directorias erradas.

Em ambiente *Cloud* o FARSITE não irá alterar o seu comportamento, continuamos sempre a necessitar do executável e do ficheiro de texto RunPanther para que

o simulador corra. Em vez de correr o simulador localmente iremos correr o simulador em ambiente *cloud*, ou seja, na instância criada, logo os procedimentos feitos localmente foram replicados na instância. Na figura 3.1.2 temos como o FARSITE corre em ambiente *Cloud*, é importante reparar que os ficheiros de entrada, pasta saída e o ficheiro RunPanther estão todos dentro da instância. É possível criar o ficheiro RunPanther e os ficheiros de entrada localmente, no entanto para correr a simulação na instância é absolutamente necessário que estes ficheiros se encontrem dentro da instância nas directorias escritas no ficheiro RunPanther.

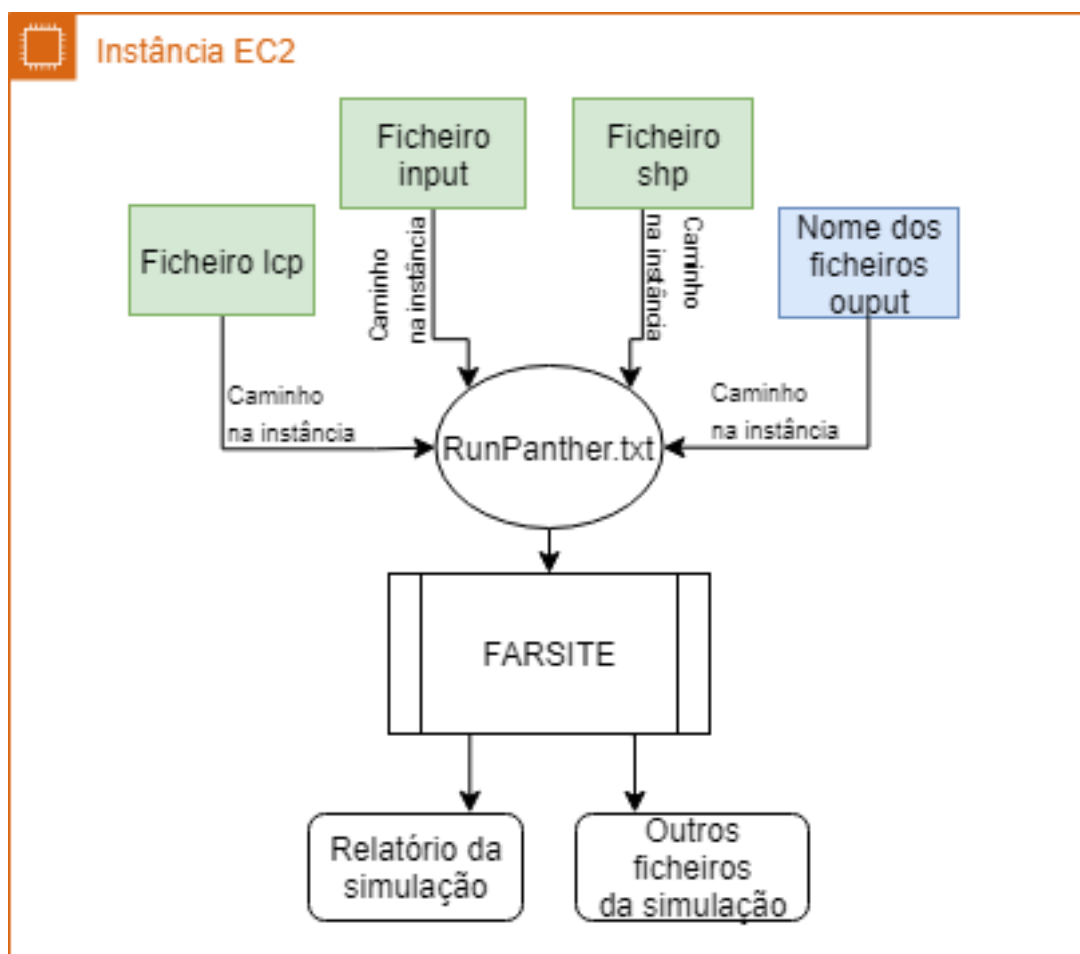


Figura 3.1.2 - Diagrama da simulação do Farsite na instância EC2.

3.2 Interface gráfica do utilizador

Nesta secção iremos tratar de como o ambiente *cloud* e o FARSITE comunicam com o utilizador final. Foi necessário criar uma interface gráfica porque nem o FARSITE nem a instância contêm uma interface deste género e assim conseguimos evitar que o utilizador final tenha de saber como o simulador foi instalado na instância e em que directorias, evitando assim a escrita de comandos por parte do utilizador final. Embora a instância deixe instalar uma interface gráfica de nada irá ajudar o utilizador final, pois o utilizador tinha de se ligar via VNC ou outro programa idêntico de maneira a poder aceder à instância. Uma vez ligado à instância o utilizador final seria obrigado a saber onde estão as pastas que necessita para a simulação e escrever as linhas de comando para correr o FARSITE.

Deste modo foi criado uma interface gráfica de utilizador de maneira a simplificar e agilizar o processo de correr o simulador e possibilitando a obtenção dos resultados de simulação.

3.2.1 Web App e Interface gráfica

Utilizou-se uma web app como interface gráfica de utilizador, onde o utilizador final poderá aceder de qualquer dispositivo desde que exista uma ligação à internet. A web app é inicializada através do *Tomcat* que foi instalado previamente na instância, o *Tomcat* é um web server que suporta linguagem java podendo assim criar uma web app na qual a interface gráfica irá ser colocada. A interface gráfica foi construída com recurso ao PrimeFaces que é uma biblioteca de componentes de interface gráfica de utilizador com código aberto para aplicações em Java.

Para acedermos à web app é necessário saber o DNS *publico* (IPv4) da instância, a porta na qual o *Tomcat* foi inicializado que por definição é a porta 8080 e saber o nome da aplicação que queremos aceder neste caso será SimuladorFarsite.

Uma vez dentro da web app o utilizador final, irá ter quatro separadores onde poderá fazer determinadas operações, nas quais poderá realizar upload de ficheiros

de entrada e ficheiros RunPanther, correr o simulador e descarregar ficheiros referentes às simulações realizadas.

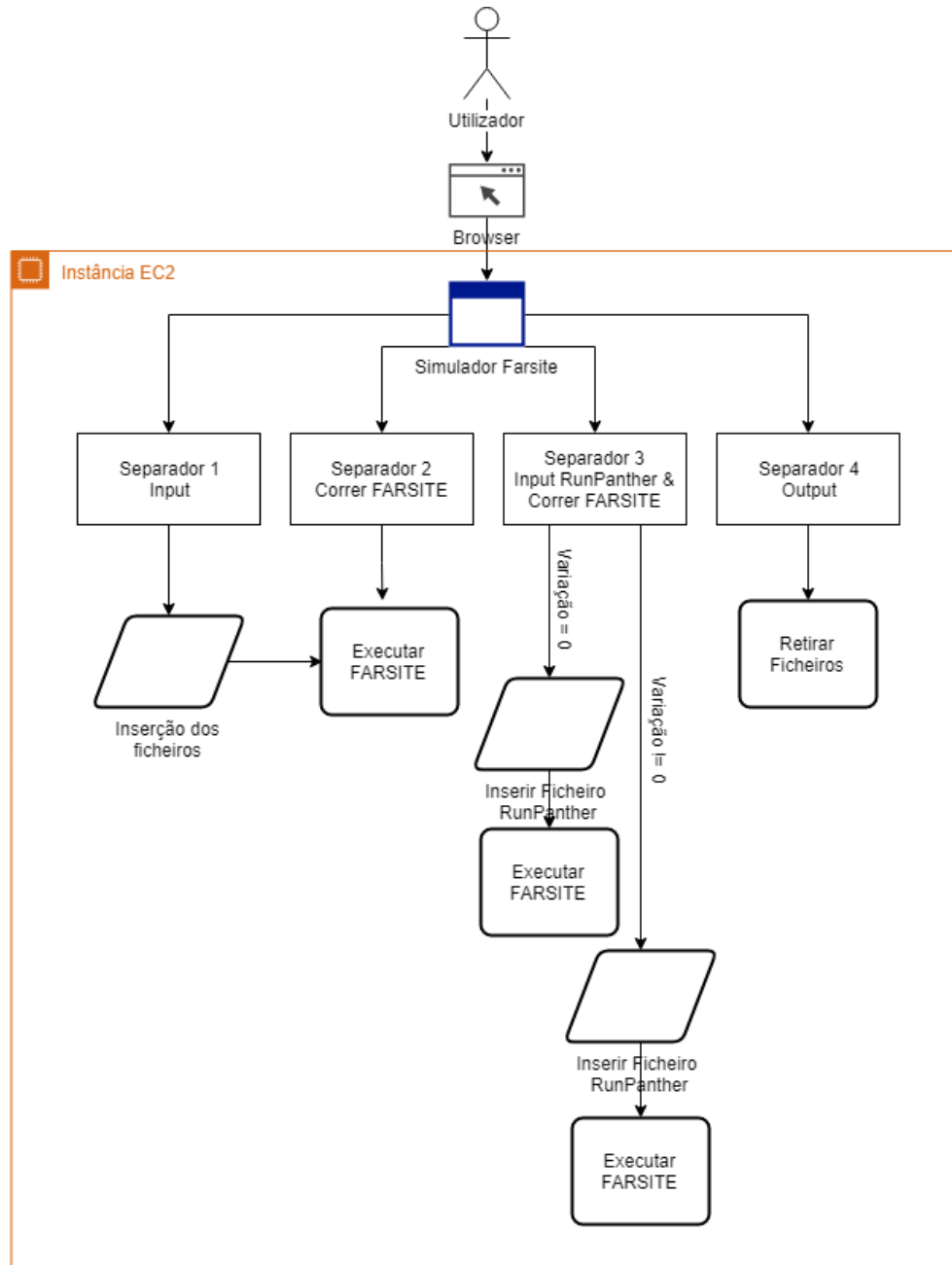


Figura 3.2.1 - Case use da Web App.

No primeiro separador, Figura 3.2.2, o utilizador poderá colocar os ficheiros de entrada (lcp, shp, input) para upload na instância, estes ficheiros serão guardados

numa pasta específica dentro da instância. Existem quatro locais para colocar os ficheiros, cada local corresponde a um tipo de ficheiro diferente, para correr o simulador é obrigatório que estes ficheiros se encontrem na instância, embora o tipo de ficheiro shx não entre no ficheiro de execução este é importante por ser um complemento do ficheiro shp. O utilizador não poderá colocar mais que um ficheiro de cada vez, e caso coloque o ficheiro num local errado a web app não aceitará o ficheiro e vai lançar um pop up de aviso.

SimuladorFarsite

Input Correr Farsite com inputs Input RunPanther Output

Neste separador poderá colocar os ficheiros necessários para as simulações.

Lcp	<div>+ Choose</div> <div></div>
Input	<div>+ Choose</div> <div></div>
Shp	<div>+ Choose</div> <div></div>
Shx	<div>+ Choose</div> <div></div>

Figura 3.2.2 - Web App separador um.

Existe um botão no segundo separador, Figura 3.2.3, que quando carregado cria um ficheiro RunPanther com as entradas inseridas no primeiro separador. Após a criação do ficheiro este é executado pelo FARSITE, esta execução só se realiza uma vez, isto só acontece, pois, o ficheiro RunPanther contém apenas uma linha de execução. É importante referir, que este comando só será executado depois de feito o upload dos ficheiros no separador um, na instância, de outra forma não irá ter o efeito desejado.

SimuladorFarsite

Neste separador pode correr o FARSITE uma vez com os inputs colocados no separador Input.

Figura 3.2.3 - Web App separador dois.

No terceiro separador, Figura 3.2.4 é possível correr o simulador desde que o utilizador coloque um ficheiro RunPanther, o utilizador tem que garantir que os ficheiros de entrada tenham sido colocados previamente na instância. Neste separador se o utilizador colocar um ficheiro RunPanther com duas linhas de entrada e saídas o simulador irá correr em paralelo as mesmas duas linhas. A web app irá dividir esse ficheiro consoante o número de linhas escritas, criando um ficheiro RunPanther novo por cada linha existente no ficheiro original e criando threads referentes a cada novo ficheiro e realizando a simulação. O simulador não irá correr sequencialmente o ficheiro original, ou seja, após simular uma linha passa para a próxima, passando assim a usar os ficheiros que contêm uma linha do ficheiro original para poder correr em paralelo.

Neste separador encontramos outro campo que se chama variação, se o utilizador pretender colocar variação esta tem de ser fornecida primeiro e só depois colocar o ficheiro RunPanther. A web app ao verificar que existe variação irá ao ficheiro RunPanther colocado pelo utilizador procurar pelo ficheiro de formato input, e criar dois ficheiros com o mesmo formato, iguais ao original onde as únicas alterações iram ser referentes à variação máxima e mínima da velocidade do vento. Estes dois ficheiros de formato input irão ser adicionados ao ficheiro RunPanther original, este processo é realizado em todas as linhas do ficheiro RunPanther, uma vez concluído este processo a web app trata este novo ficheiro RunPanther como tendo sido introduzido sem variação.

SimuladorFarsite

Neste separador poderá correr o FARSITE x vezes consoante o numero de linhas colocadas no ficheiro runPanther.txt aqui inserido.

Poderá colocar variação na velocidade do vento se pretender.

Var(%)

Figura 3.2.4 - Web App separador três.

Os valores de variação máxima e mínima são calculados da seguinte maneira:

- $VariaçãoMaxima = ValorOriginal \times \left(1 + \frac{Variação}{100}\right)$
- $VariaçãoMinima = ValorOriginal \times \left(1 - \frac{Variação}{100}\right)$

Por último, temos o separador do output onde o utilizador terá de introduzir o nome do ficheiro de saída com a respectiva extensão que pretende retirar da instância, e a web app irá descarregar o ficheiro pretendido para o dispositivo local. É importante referir que os ficheiros estão a ser retirados da pasta de output com a seguinte directoria, /home/ubuntu/examples/Panther/cust/ouput/.

SimuladorFarsite

Irá tirar um ficheiro de texto com um relatório sobre a simulação(incluir extensão)

Figura 3.2.5 - Web App separador quatro.

A implementação encontra-se concluída, tendo-se em funcionamento um simulador de incêndio em ambiente *cloud* capaz de executar simulações. De forma a colmatar a inexistência de uma interface gráfica ligada a esse mesmo simulador e pretendendo facilitar a interação do utilizador com o simulador, foi criada uma web app. A web app tem as funções descritas na secção anterior que possibilitam uma interação do utilizador com o simulador mais eficiente e concisa. Através da instância e com recurso à web app o FARSITE é agora capaz de funcionar em paralelo e alterar os ficheiros de entrada proporcionando um melhor estudo do incêndio e não necessitando de recorrer a vários ficheiros de simulação.

4

Resultados

O capítulo dos resultados é onde fazemos os testes à implementação realizada e verificamos se existe algum problema ou se está tudo a funcionar como desejado. Pretendemos verificar se a instância é capaz de suportar com o processamento em paralelo do simulador e qual o máximo de simulações em paralelo. A web app necessita de inserir bem os ficheiros de entrada e de os retirar, ser capaz de criar os ficheiros de execução do simulador e também criar a variação no ficheiro.

Começamos primeiro por ver se a web app era capaz de inserir todos os tipos de ficheiros que são necessários para o funcionamento do simulador e se estes ficheiros são colocados nos locais respectivos.

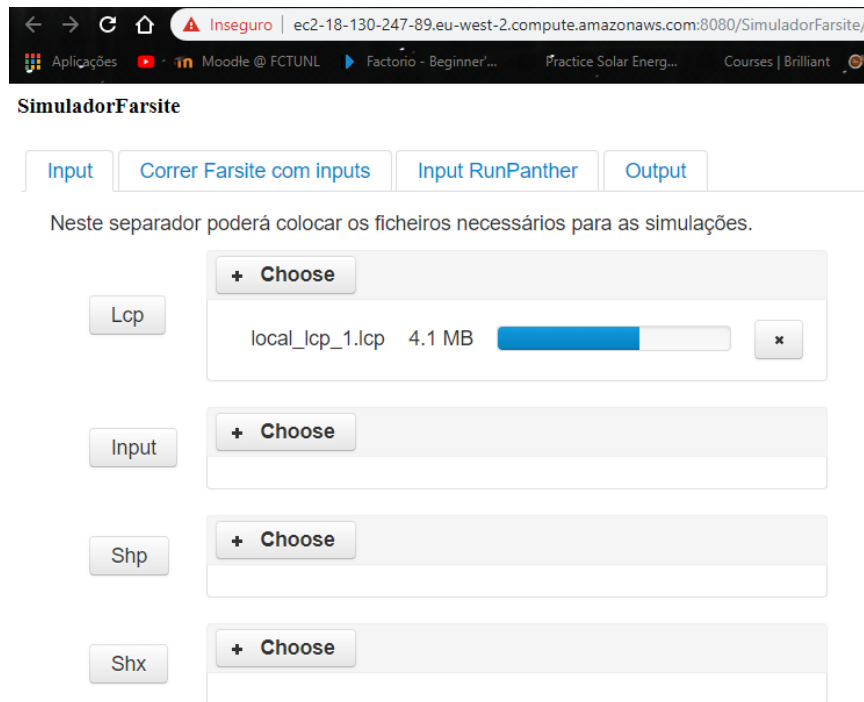


Figura 4.1 - Web app, colocação de um tipo de ficheiro na instância.

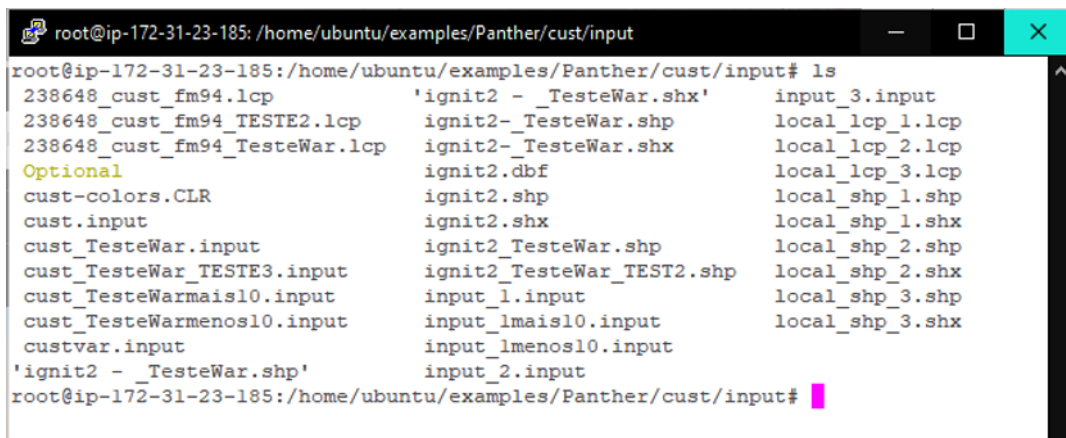


Figura 4.2 - Putty, confirmação da colocação dos ficheiros na instância.

A figura 4.1 em cima representa um exemplo de como os ficheiros de entrada são colocados na instância através da web app. Uma vez inserido o ficheiro pretendido foi necessário confirmar se o ficheiro estava a ser colocado no local designado, usou-se o Putty para esta confirmação representado na figura 4.2.

Durante os testes correspondentes à colocação de ficheiros na web app, não existiu qualquer tipo de falhas, todos os ficheiros foram enviados com sucesso e para a pasta respectiva na instância não demorando mais que 5 segundos, com a excepção do ficheiro lcp devido ao seu tamanho ser consideravelmente maior que o dos outros ficheiros.

Uma vez que o separador um que trata da inserção dos ficheiros se encontra a funcionar sem qualquer tipo de problemas podemos passar para os testes do separador dois. Neste separador é criado um ficheiro RunPanther com recurso aos ficheiros colocados no primeiro separador, este executa apenas uma simulação. Esta simulação cria um ficheiro RunPanther gerado a partir dos últimos três tipos de ficheiros inseridos (lcp, input e shp) no primeiro separador, ou seja, caso insira um ficheiro lcp com o nome lcp_1 e de seguida coloque outro com o nome lcp_2 o ficheiro RunPanther é criado com o ficheiro lcp mais recente, no entanto os ficheiros colocados anteriormente ficaram guardados na instância e se colocar outro ficheiro, com o mesmo nome que outro, já existente este é reescrito com a informação do último. Isto acontece com todos os formatos de ficheiros que se encontram no separador um, é necessário ter em atenção os ficheiros shp e shx pois estes formam um par e se ambos não forem inseridos com o mesmo nome ou se não colocar um deles o FARSITE não irá correr. Após uma inserção correcta dos ficheiros, o simulador irá correr uma vez e as suas saídas terão o seguinte formato: outputAAAAMMDD-HHMM, na figura 4.3 tem um exemplo de uma execução bem-sucedida do simulador.

```

output20190914-1513_ArrivalTime.asc      output20190914-1513_Perimeters.shx
output20190914-1513_ArrivalTime.fbg      output20190914-1513_ReactionIntensity.asc
output20190914-1513_CrownFire.asc        output20190914-1513_ReactionIntensity.fbg
output20190914-1513_CrownFire.fbg        output20190914-1513_SpotGrid.asc
output20190914-1513_FlameLength.asc       output20190914-1513_Spots.csv
output20190914-1513_FlameLength.fbg      output20190914-1513_Spots.dbf
output20190914-1513_HeatPerUnitArea.asc   output20190914-1513_Spots.shp
output20190914-1513_HeatPerUnitArea.fbg   output20190914-1513_Spots.shx
output20190914-1513_Ignitions.asc         output20190914-1513_SpreadDirection.asc
output20190914-1513_Intensity.asc         output20190914-1513_SpreadDirection.fbg
output20190914-1513_Intensity.fbg        output20190914-1513_SpreadRate.asc
output20190914-1513_Perimeters.dbf        output20190914-1513_SpreadRate.fbg
output20190914-1513_Perimeters.shp       output20190914-1513_Timings.txt

```

Figura 4.3 - Putty, formato do output da simulação bem sucedida no segundo separador da web app.

Com a simulação bem-sucedida da figura anterior e através do ficheiro de saída *output20190914-1513_Timings.txt* podemos verificar se a simulação foi realizada com as entradas que o utilizador pretendia, e é possível ver neste ficheiro o tempo no qual o FARSITE realizou a simulação e quais as entradas utilizadas nessa mesma simulação. Na figura 4.4 em baixo temos o tempo que o simulador demorou a correr e este tempo coincide com o tempo que os ficheiros de saída ficaram disponíveis na instância para visualização.

```

5 6 400 6 120 62
5 6 500 2 210 63
5 6 600 2 210 63
5 6 700 2 210 63
5 6 800 2 210 63
5 6 900 2 210 63
5 6 1000 2 210 63
5 6 1100 8 230 66
5 6 1200 8 230 66
5 6 1300 8 230 66
5 6 1400 8 230 66
5 6 1500 8 230 66
5 6 1600 8 230 66
5 6 1700 7 220 65
5 6 1800 7 220 65
5 6 1900 7 220 65
5 6 2000 7 220 65
5 7 200 7 220 65

Total Farsite Run Time: 38.560 Seconds

```

Figura 4.4 - Putty, visualização do tempo de resposta no ficheiro de saída Timings.

No próximo separador o utilizador poderá colocar um ficheiro RunPanther que tem a possibilidade de conter mais que uma linha, consultar anexo 7.3 figura 7.3.1. Na figura 4.5 em baixo está uma simulação de um ficheiro RunPanther de 3 linhas,

nas quais foram divididas pela web app e executadas em paralelo pelo FARSITE. Cada linha tem um conjunto de saídas com prefixos diferentes, isto acontece quando o ficheiro RunPanther é criado pelo utilizador. Este tem de garantir que cada nome de saída tenha de ser diferente, caso isto não aconteça o simulador irá sobrepor a simulação com outra, prevalecendo a que acabou mais tarde.

```

root@ip-172-31-23-185: /home/ubuntu/examples/Panther/cust/output
root@ip-172-31-23-185: /home/ubuntu/examples/Panther/cust/output# ls
output1_runpanther3_ArrivalTime.asc      output20190914-1513_Intensity.asc      output2_runpanther3_Spots.dbf
output1_runpanther3_ArrivalTime.fbg      output20190914-1513_Intensity.fbg      output2_runpanther3_Spots.shp
output1_runpanther3_CrownFire.asc        output20190914-1513_Perimeters.dbf     output2_runpanther3_Spots.shx
output1_runpanther3_CrownFire.fbg        output20190914-1513_Perimeters.shp     output2_runpanther3_SpreadDirection.asc
output1_runpanther3_FlameLength.asc       output20190914-1513_Perimeters.shx     output2_runpanther3_SpreadDirection.fbg
output1_runpanther3_FlameLength.fbg       output20190914-1513_ReactionIntensity.asc output2_runpanther3_SpreadRate.asc
output1_runpanther3_HeatPerUnitArea.asc   output20190914-1513_ReactionIntensity.fbg output2_runpanther3_SpreadRate.fbg
output1_runpanther3_HeatPerUnitArea.fbg   output20190914-1513_SpotGrid.asc       output2_runpanther3_Timings.txt
output1_runpanther3_Ignitions.asc         output20190914-1513_Spots.csv          output3_runpanther3_ArrivalTime.asc
output1_runpanther3_Intensity.asc        output20190914-1513_Spots.dbf         output3_runpanther3_ArrivalTime.fbg
output1_runpanther3_Intensity.fbg        output20190914-1513_Spots.shp         output3_runpanther3_CrownFire.asc
output1_runpanther3_Perimeters.dbf       output20190914-1513_Spots.shx         output3_runpanther3_CrownFire.fbg
output1_runpanther3_Perimeters.shp       output20190914-1513_SpreadDirection.asc output3_runpanther3_FlameLength.asc
output1_runpanther3_Perimeters.shx       output20190914-1513_SpreadDirection.fbg output3_runpanther3_FlameLength.fbg
output1_runpanther3_ReactionIntensity.asc output20190914-1513_SpreadRate.asc      output3_runpanther3_HeatPerUnitArea.asc
output1_runpanther3_ReactionIntensity.fbg output20190914-1513_SpreadRate.fbg     output3_runpanther3_HeatPerUnitArea.fbg
output1_runpanther3_SpotGrid.asc         output20190914-1513_Timings.txt       output3_runpanther3_Ignitions.asc
output1_runpanther3_Spots.csv            output2_runpanther3_ArrivalTime.asc    output3_runpanther3_Intensity.asc
output1_runpanther3_Spots.dbf            output2_runpanther3_ArrivalTime.fbg   output3_runpanther3_Intensity.fbg
output1_runpanther3_Spots.shp            output2_runpanther3_CrownFire.asc     output3_runpanther3_Perimeters.dbf
output1_runpanther3_Spots.shx            output2_runpanther3_CrownFire.fbg     output3_runpanther3_Perimeters.shp
output1_runpanther3_SpreadDirection.asc  output2_runpanther3_FlameLength.asc   output3_runpanther3_Perimeters.shx
output1_runpanther3_SpreadDirection.fbg  output2_runpanther3_FlameLength.fbg   output3_runpanther3_ReactionIntensity.asc
output1_runpanther3_SpreadRate.asc       output2_runpanther3_HeatPerUnitArea.asc output3_runpanther3_ReactionIntensity.fbg
output1_runpanther3_SpreadRate.fbg       output2_runpanther3_HeatPerUnitArea.fbg output3_runpanther3_SpotGrid.asc
output1_runpanther3_Timings.txt          output2_runpanther3_Ignitions.asc     output3_runpanther3_Spots.csv
output20190914-1513_ArrivalTime.asc     output2_runpanther3_Intensity.asc     output3_runpanther3_Spots.dbf
output20190914-1513_ArrivalTime.fbg     output2_runpanther3_Intensity.fbg     output3_runpanther3_Spots.shp
output20190914-1513_CrownFire.asc       output2_runpanther3_Perimeters.dbf    output3_runpanther3_Spots.shx
output20190914-1513_CrownFire.fbg       output2_runpanther3_Perimeters.shp    output3_runpanther3_SpreadDirection.asc
output20190914-1513_FlameLength.asc     output2_runpanther3_Perimeters.shx    output3_runpanther3_SpreadDirection.fbg
output20190914-1513_FlameLength.fbg     output2_runpanther3_ReactionIntensity.asc output3_runpanther3_SpreadRate.asc
output20190914-1513_HeatPerUnitArea.asc output2_runpanther3_ReactionIntensity.fbg output3_runpanther3_SpreadRate.fbg
output20190914-1513_HeatPerUnitArea.fbg output2_runpanther3_SpotGrid.asc      output3_runpanther3_Timings.txt
output20190914-1513_Ignitions.asc       output2_runpanther3_Spots.csv

```

Figura 4.5 - Putty, simulação em paralelo com 3 linhas sem variação.

Este separador ainda tem outra funcionalidade, é possível colocar um valor de variação na velocidade do vento que se encontra no ficheiro de entrada com formato input. A figura 4.6, em baixo, é uma parte de dois ficheiros de formato input, onde mostra as colunas onde está informação referente ao vento, a velocidade do vento está na coluna 8, é esta coluna que entra para as contas do cálculo da variação máxima e mínima descrita na implementação.

RAWS: 249													RAWS: 249												
2013	4	26	1200	77	26	0.00	4	220	25	2013	4	26	1200	77	26	0.00	4.24	220	25						
2013	4	26	1300	79	23	0.00	4	225	32	2013	4	26	1300	79	23	0.00	4.24	225	32						
2013	4	26	1400	78	21	0.00	4	290	39	2013	4	26	1400	78	21	0.00	4.24	290	39						
2013	4	26	1500	78	19	0.00	4	316	47	2013	4	26	1500	78	19	0.00	4.24	316	47						
2013	4	26	1600	77	21	0.00	3	273	54	2013	4	26	1600	77	21	0.00	3.18	273	54						
2013	4	26	1700	74	23	0.00	2	267	61	2013	4	26	1700	74	23	0.00	2.12	267	61						
2013	4	26	1800	64	40	0.00	0	65	69	2013	4	26	1800	64	40	0.00	0.00	65	69						
2013	4	26	1900	60	52	0.00	1	84	70	2013	4	26	1900	60	52	0.00	1.06	84	70						
2013	4	26	2000	60	52	0.00	0	80	72	2013	4	26	2000	60	52	0.00	0.00	80	72						
2013	4	26	2100	61	38	0.00	0	53	73	2013	4	26	2100	61	38	0.00	0.00	53	73						
2013	4	26	2200	59	36	0.00	0	45	75	2013	4	26	2200	59	36	0.00	0.00	45	75						
2013	4	26	2300	59	38	0.00	1	61	76	2013	4	26	2300	59	38	0.00	1.06	61	76						
2013	4	27	0000	59	30	0.00	1	17	78	2013	4	27	0000	59	30	0.00	1.06	17	78						
2013	4	27	0100	58	33	0.00	1	117	80	2013	4	27	0100	58	33	0.00	1.06	117	80						
2013	4	27	0200	58	31	0.00	1	264	81	2013	4	27	0200	58	31	0.00	1.06	264	81						
2013	4	27	0300	58	37	0.00	1	82	83	2013	4	27	0300	58	37	0.00	1.06	82	83						
2013	4	27	0400	58	31	0.00	1	83	84	2013	4	27	0400	58	31	0.00	1.06	83	84						
2013	4	27	0500	57	34	0.00	1	357	86	2013	4	27	0500	57	34	0.00	1.06	357	86						
2013	4	27	0600	58	47	0.00	1	311	88	2013	4	27	0600	58	47	0.00	1.06	311	88						
2013	4	27	0700	64	43	0.00	2	244	77	2013	4	27	0700	64	43	0.00	2.12	244	77						
2013	4	27	0800	70	35	0.00	2	235	67	2013	4	27	0800	70	35	0.00	2.12	235	67						

Figura 4.6 - À esquerda valores originais da velocidade do vento e à direita valores com uma variação máxima de 6%.

Colocou-se uma variação e de seguida um ficheiro RunPanther com 3 linhas na figura 4.7. A simulação irá gerar 9 conjuntos de saídas, 3 com valores de velocidade do vento original, outros 3 com velocidade do vento com variação máxima e os 3 últimos com velocidade do vento com variação mínima. O conjunto de saídas que contém variação são os ficheiros que contêm um 0 ou 1 a seguir ao nome escrito pelo utilizador na criação do ficheiro RunPanther, o valor de variação máxima é representado pelo 1, a variação mínima pelo 0, o valor original mantém o nome escrito pelo utilizador.

O tempo de simulação é aproximadamente 40 segundos por cada linha, que é separada pela web app do ficheiro RunPanther inserido pelo utilizador, quer com ou sem variação. No entanto existe um aumento do tempo na disponibilidade dos ficheiros na instância após a simulação, quantas mais linhas o ficheiro RunPanther original tiver mais tempo o utilizador vai ter de esperar para ver os resultados da simulação. Para estas simulações correrem bem e sem qualquer tipo de problemas, o utilizador tem sempre de garantir que os ficheiros de entrada que colocou no ficheiro RunPanther estão colocados na instância utilizando o separador um.

Conclusão e trabalhos futuros

5.1 Conclusão

Os incêndios florestais são cada vez mais frequentes em todo o mundo devido a inúmeros motivos, podendo estes serem originados naturalmente ou não, com este propósito foi realizada esta dissertação. O trabalho proposto neste documento tem como objectivo provar que o conceito de que um simulador de incêndios consegue correr em ambiente *cloud* fornecendo mais uma ferramenta a todas as entidades que combatem os incêndios em Portugal.

Esta dissertação consistiu no estudo do processamento de simulações da evolução de incêndios florestais em ambiente *cloud*, foi usado um simulador de incêndio chamado FARSITE. Este programa é determinista e usa a maioria dos modelos de comportamento de incêndio estudados neste documento tornando o FARSITE o programa mais completo dos estudados no estado de arte. O estudo em ambientes *cloud* foi realizado com base no que estes ambientes podem oferecer de vantagens e desvantagens, quais modelos e serviços existentes no mercado e que fornecedores oferecem os serviços e modelos usados para a implementação desta dissertação. Foi usada a Amazon (AWS) como fornecedor por existir um nível gratuito no qual foi possível testar a implementação sem qualquer tipo de gastos desde que se tome algumas precauções.

A implementação consistiu na junção do simulador de incêndios com o ambiente *cloud*, para isso usou-se uma máquina virtual denominada instância onde o simulador correu. Devido à falta de interface gráfica de utilizador tanto do simulador como da instância foi criado em adição à implementação uma web app com o propósito de facilitar a utilização do simulador pelo utilizador final. A web app é capaz de inserir e retirar ficheiros referentes ao simulador e também de correr o simulador com ficheiros de execução denominados RunPanther. Esta web app pode correr o simulador em paralelo e criar ficheiros de entrada com variação na velocidade dos ventos.

Os resultados foram realizados com base nos separadores da web app que o utilizador final irá utilizar. Começou-se por testar o primeiro separador que trata da colocação dos ficheiros onde correu com sucesso e sem qualquer tipo de erro. Na segunda parte testou-se a criação de um ficheiro RunPanther com recursos aos ficheiros inseridos anteriormente e os resultados correram dentro do espectável. O terceiro separador no qual o utilizador final pode colocar um ficheiro de execução com ou sem variação e o simulador correr em paralelo, não foram detectados nenhuns erros, mas os ficheiros de saída demoraram a ficar disponíveis na instância, este tempo tende a aumentar consoante o número de linhas do ficheiro de execução. Por último temos o separador onde o utilizador poderá retirar ficheiros da instância referentes às simulações realizadas previamente, nestes testes não houve qualquer tipo de problema.

A implementação demonstrou-se um desafio, devido ao uso de muitas tecnologias e ferramentas novas que foi necessário aprender a usar. Um dos desafios foi o próprio ambiente *cloud* na medida em que sabia tão pouco da tecnologia, no entanto mostrou ser uma tecnologia muito versátil e interessante de trabalhar. Foi preciso ter em atenção, alguns factores durante a utilização do ambiente *cloud*, especialmente quando se insere uma web app na instância e posteriormente quando se realizam os testes necessários para a implementação da mesma, pois podemos acabar por nos tornar um utilizador não gratuito na AWS, sendo necessário apartir daí pagar por cada hora de utilização da instancia

5.2 Trabalhos futuros

Para trabalhos futuros sugiro que se continue a melhorar a web app de maneira a ter mais funcionalidades, tais como:

- O utilizador possa criar os seus ficheiros de execução na web app, evitando assim erros que este possa cometer.
- Onde a variação possa ser mais genérica, ou seja, poder-se colocar vários tipos de variação para várias entradas.

Referente à parte do ambiente *cloud*, devido a ter sido utilizador num nível gratuito de acesso á mesma, sugeria que se repetissem estes mesmos testes com uma instância com mais capacidade de processamento e memória RAM.

6

Bibliografia

- [1] "A MATHEMATICAL MODEL FOR PREDICTING FIRE SPREAD."
- [2] C. E. Van Wagner, "Conditions for the start and spread of crown fire," *Can. J. For. Res.*, vol. 7, no. 1, pp. 23–34, 1977.
- [3] F. Service and G. A. Morris, "A Simple Method for Computing Spotting Distances From Surface Fires," 1986.
- [4] F. Service and R. C. Rothermel, "How to Predict the Spread and Intensity of Forest and Range Fires," 1983.
- [5] F. Service and R. C. Rothermel, "Predicting Behavior and Size of Crown Fires in the Northern Rocky Mountains," no. January, 1991.
- [6] "BehavePlus | Fire, Fuel, and Smoke Science Program." [Online]. Available: <https://www.firelab.org/project/behaveplus>. [Accessed: 06-Feb-2019].
- [7] "Fire Characteristics Chart | Fire, Fuel, and Smoke Science Program." [Online]. Available: <https://www.firelab.org/project/fire-characteristics-chart>. [Accessed: 06-Feb-2019].
- [8] "FARSITE | Fire, Fuel, and Smoke Science Program." [Online]. Available: <https://www.firelab.org/project/farsite>. [Accessed: 06-Feb-2019].
- [9] R. Buyya, C. S. Yeo, S. Venugopal, J. Broberg, and I. Brandic, "Cloud computing and emerging IT platforms: Vision, hype, and reality for delivering computing as the 5th utility," *Futur. Gener. Comput. Syst.*, vol. 25, no. 6, pp. 599–616, 2009.
- [10] Omniblend, "National Institute of Standards and Technology," 2011.
- [11] "What is cloud computing? A beginner's guide | Microsoft Azure." [Online].

- Available: <https://azure.microsoft.com/en-in/overview/what-is-cloud-computing/>. [Accessed: 29-Jan-2019].
- [12] S. Goyal, "Public vs Private vs Hybrid vs Community - Cloud Computing: A Critical Review," *Int. J. Comput. Netw. Inf. Secur.*, vol. 6, no. 3, pp. 20–29, 2014.
 - [13] "What is *cloud* computing? - Definition from WhatIs.com." [Online]. Available: <https://searchcloudcomputing.techtarget.com/definition/cloud-computing>. [Accessed: 29-Jan-2019].
 - [14] "Exemplos de *cloud*." [Online]. Available: <https://www.quora.com/What-is-a-public-cloud-with-an-example>. [Accessed: 30-Jan-2019].
 - [15] C. Rong, S. T. Nguyen, and M. G. Jaatun, "Beyond lightning: A survey on security challenges in *cloud* computing," *Comput. Electr. Eng.*, vol. 39, no. 1, pp. 47–54, 2013.
 - [16] "Private, Hybrid and Public Cloud Explained e Exemplos." [Online]. Available: <https://www.business2community.com/cloud-computing/Private-hybrid-public-cloud-explained-0789846>. [Accessed: 30-Jan-2019].
 - [17] "Exemplos de *Cloud* 2." [Online]. Available: <https://www.technopulse.com/2011/10/cloud-deployment-Private-public-example.html>. [Accessed: 30-Jan-2019].
 - [18] "Explaining the *Community Cloud* | Data Center Knowledge." [Online]. Available: <https://www.datacenterknowledge.com/archives/2014/10/13/explaining-community-cloud>. [Accessed: 29-Jan-2019].
 - [19] "What is *cloud* service provider (*cloud* provider)? - Definition from WhatIs.com." [Online]. Available: <https://searchitchannel.techtarget.com/definition/cloud-service-provider-cloud-provider>. [Accessed: 29-Jan-2019].
 - [20] M. A. Nadeem, "Cloud Computing: Security Issues and Challenges," *J. Wirel. Commun.*, vol. 1, no. 1, 2016.
 - [21] "What are *Cloud* Service Providers? - Definition - SDxCentral .com." [Online]. Available: <https://www.sdxcentral.com/cloud/definitions/what-are-cloud-service-providers/>. [Accessed: 29-Jan-2019].



Anexos

7.1 Inicialização Instância na AWS

Foi usado o serviço EC2 da AWS, o serviço EC2 é a criação de uma máquina virtual de qualquer sistema operativo que o utilizador pretende este serviço é denominado instância. A instância deixa escolher uma variedade de opções conforme aquilo que o utilizador necessite desde o tipo de instância, a configuração da instância, armazenamento e configuração do sistema de segurança, algumas destas opções são limitadas para utilizadores gratuitos.

Na figura 7.1.1, em baixo está uma visualização geral de como uma instância corre em ambiente *Cloud* na AWS e como é possível o utilizador aceder à mesma através de qualquer computador desde que tenha a chave privada.

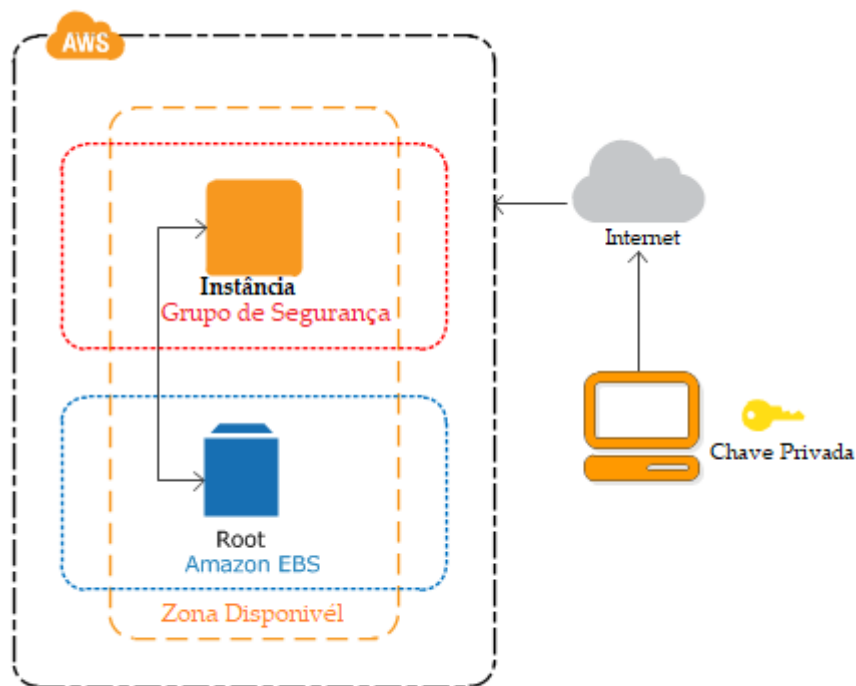


Figura 7.1.1 - Visualização geral de uma instância EC2.

Ao criar uma instância na AWS é necessário passar certas etapas para podermos ter a instância com a configuração desejada, as próximas figuras são a representação dessas mesmas etapas e as configurações para a implementação.

Na figura 7.1.2, em baixo é onde escolhemos o sistema operativo no qual queremos que a nossa instância corra. Optamos pela instância " *Ubuntu Server 18.04 LTS (HVM), SSD Volume Type - ami-077a5b1762a2dde35* " por ser compatível com Java e com #C e por ser um do sistema operativo gratuito.

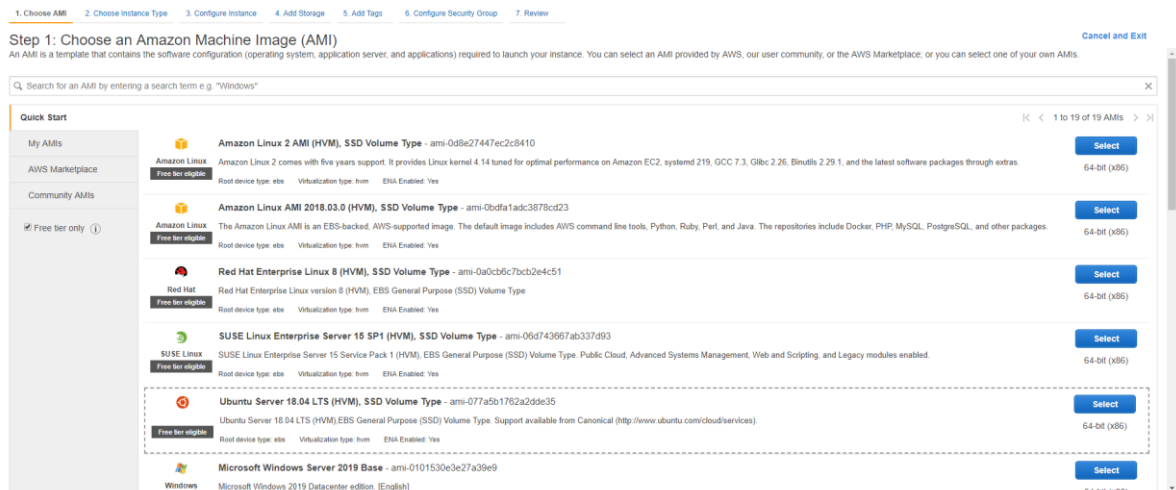


Figura 7.1.2 - EC2, primeiro passo escolha do sistema operativo.

Na figura 7.1.3, em baixo podemos ver todas as opções de hardware que temos a nossa disposição para escolher para a nossa máquina virtual. Existe várias combinações entre CPU, memória, armazenamento e conectividade, escolheu-se "*t2.micro* - *Variable ECUs*, *1 vCPUs*, *2.5 GHz*, *Intel Xeon Family*, *1 GiB memory*, *EBS only*" pois é a única fornecida para utilizadores gratuitos.

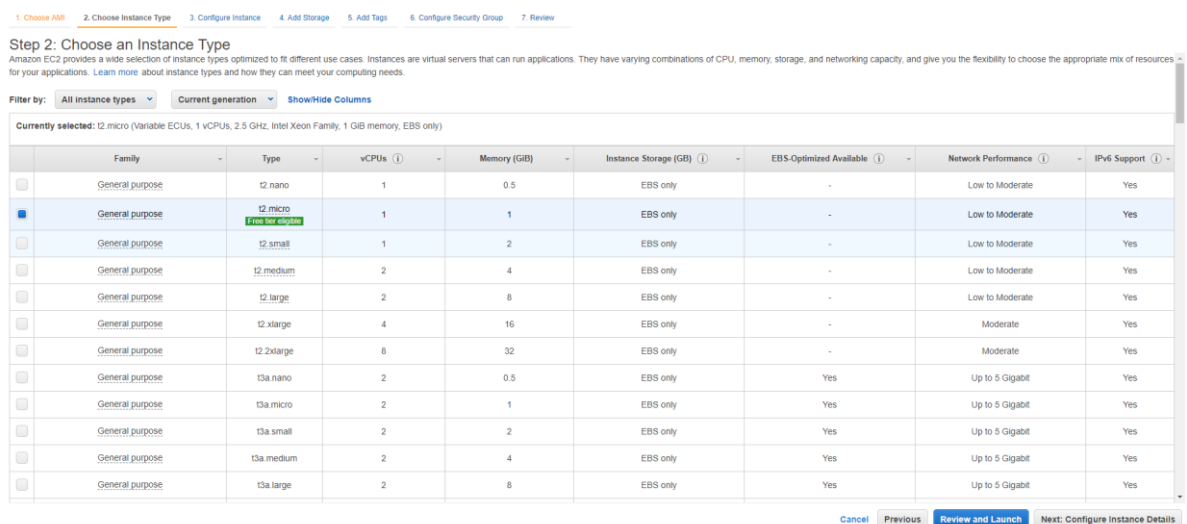


Figura 7.1.3 - EC2, segundo passo escolha do tipo de instância.

A próxima etapa é onde podemos configurar a nossa instância com os requisitos mais avançados, tal como o número de instâncias a lançar, conectividade entre outros. Na figura 7.1.4 mostra as configurações padrão dadas pela AWS e foram estas que optamos para podermos continuar na categoria de utilizador gratuito.

Step 3: Configure Instance Details
Configure the instance to suit your requirements. You can launch multiple instances from the same AMI, request Spot instances to take advantage of the lower pricing, assign an access management role to the instance, and more.

Number of instances 1 [Launch into Auto Scaling Group](#)

Purchasing option ☐ Request Spot Instances

Network vpc-65b7u70r (default) [Create new VPC](#)

Subnet No preference (default subnet in any Availability Zone) [Create new subnet](#)

Auto-assign Public IP Use subnet setting (Enable)

Placement group ☐ Add instance to placement group

Capacity Reservation Open [Create new Capacity Reservation](#)

IAM role None [Create new IAM role](#)

Shutdown behavior Stop

Enable termination protection ☐ Protect against accidental termination

Monitoring ☐ Enable CloudWatch detailed monitoring
Additional charges apply

Tenancy Shared - Run a shared hardware instance
Additional charges will apply for dedicated tenancy

T2/T3 Unlimited ☐ Enable
Additional charges may apply

► Advanced Details

[Cancel](#) [Previous](#) [Review and Launch](#) [Next: Add Storage](#)

Figura 7.1.4 - EC2, terceiro passo escolha de configurações avançadas.

A seguinte figura 7.1.5 mostra o tipo de armazenamento que podemos optar bem como quanto espaço podemos alocar para a nossa instância. Não esquecendo que o utilizador gratuito pode alocar 30Gb para cada instância, para a instância que iremos usar alocamos 8Gb com grande margem de crescimento.

Step 4: Add Storage
Your instance will be launched with the following storage device settings. You can attach additional EBS volumes and instance store volumes to your instance, or edit the settings of the root volume. You can also attach additional EBS volumes after launching an instance, but not instance store volumes. [Learn more](#) about storage options in Amazon EC2.

Volume Type	Device	Snapshot	Size (GiB)	Volume Type	IOPS	Throughput (MiB/s)	Delete on Termination	Encryption
Root	/dev/sda1	snap-090bfa34796733b92	8	General Purpose SSD (gp2)	100 / 3000	N/A	<input checked="" type="checkbox"/>	Not Encrypted

[Add New Volume](#)

Free tier eligible customers can get up to 30 GB of EBS General Purpose (SSD) or Magnetic storage. [Learn more](#) about free usage tier eligibility and usage restrictions.

Figura 7.1.5 - EC2, quarto passo alocação de memória.

O último passo a ser realizado é o da configuração da segurança, é neste passo que indicamos quais os protocolos de segurança que a nossa instância irá recorrer quando for inicializada e estiver a funcionar, figura 7.1.6.

Figura 7.1.6 - EC2, quinto passo configuração de segurança.

No fim a AWS pede para fazer uma revisão das opções que colocou para depois avançar com o lançamento da instância em ambiente *cloud*, neste momento a AWS pede para criar uma chave encriptada denominada "key pem", esta mesma chave irá ser usada para aceder à instância em qualquer lado. Na tabela 7.1.1 estão alguns nomes de utilizador padrão fornecidos pela AWS no acesso à instância.

Tabela 7.1.1 - EC2 nomes de utilizadores padrão para acesso de SSH.

Sistema operativo	Nome de utilizador padrão para SSH
Ubuntu	Ubuntu
Linux, Windows, RHEL 5	Ec2-user
Debian	Debian

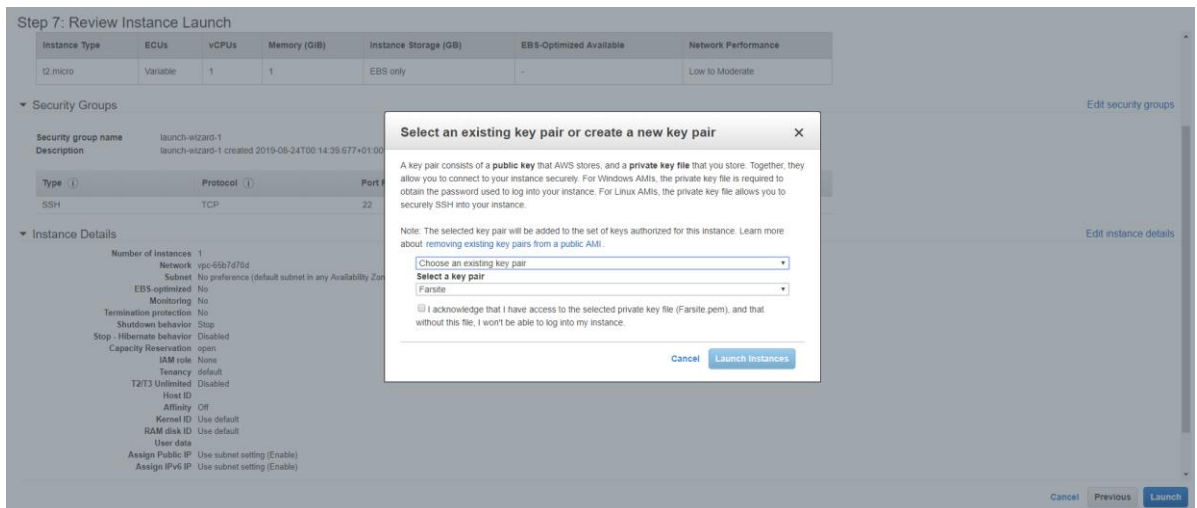


Figura 7.1.7 - EC2 conclusão da configuração da instância e criação do ficheiro pem

7.2 Aceder à instância via Putty

Tendo a instância criada na AWS, foi necessário aceder à instância para podermos instalar novas actualizações da máquina virtual e podermos instalar o Java e o *Tomcat*, foi usado o programa Putty que é um programa de emulação de terminal capaz trabalhar com o SSH, garantindo assim uma ligação segura à instância. Para estabelecer a ligação entre o Putty e a instância é preciso três coisas, o DNS *Publico* (IPv4) da instância, o ficheiro pem (chave encriptada) que é criado ao finalizar uma instância e o nome de utilizador padrão para a instância criada correspondente. Através da chave encriptada o Putty cria um ficheiro de extensão ppk no qual irá guarda a chave encriptada fornecida, este ficheiro é gerado por simplicidade para o utilizador não necessite de inserir a chave encriptada sempre que quer ligar se à instância.

Tendo o ficheiro ppk é possível criar uma sessão no Putty onde precisamos de colocar o ficheiro ppk e o DNS *publico* da instância para nos ligarmos ao terminal da instância, figura 7.2.1 e figura 7.2.2.

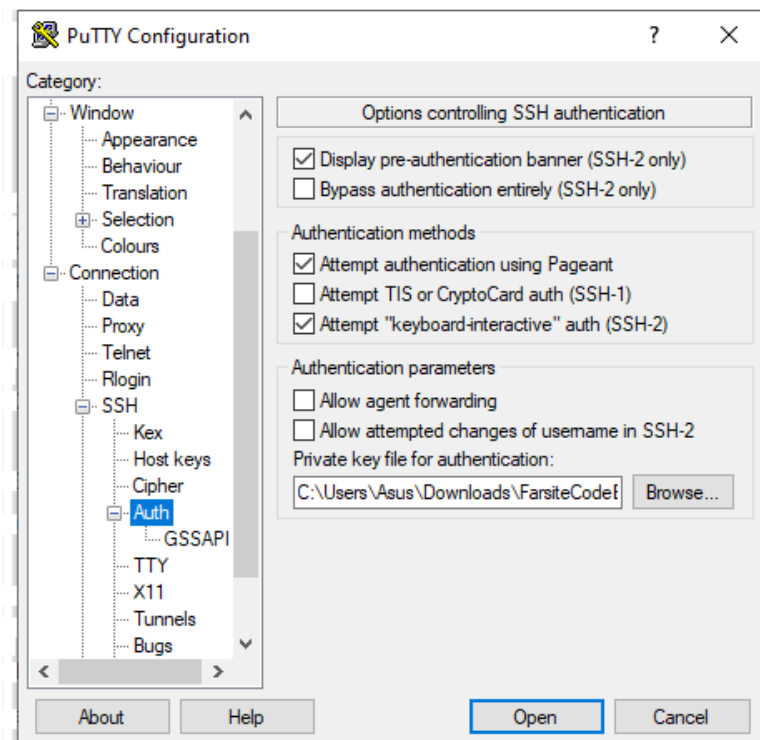


Figura 7.2.1 - Putty colocação do ficheiro ppk

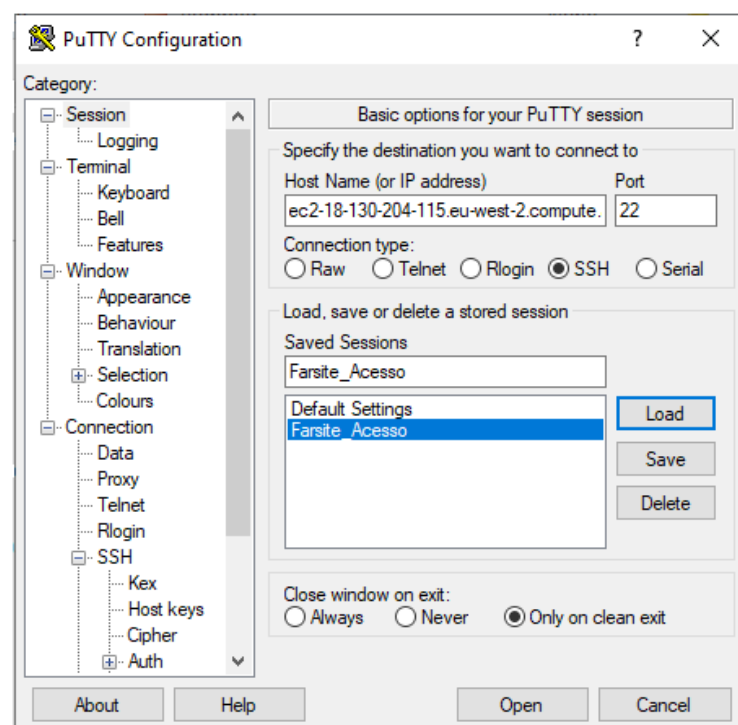
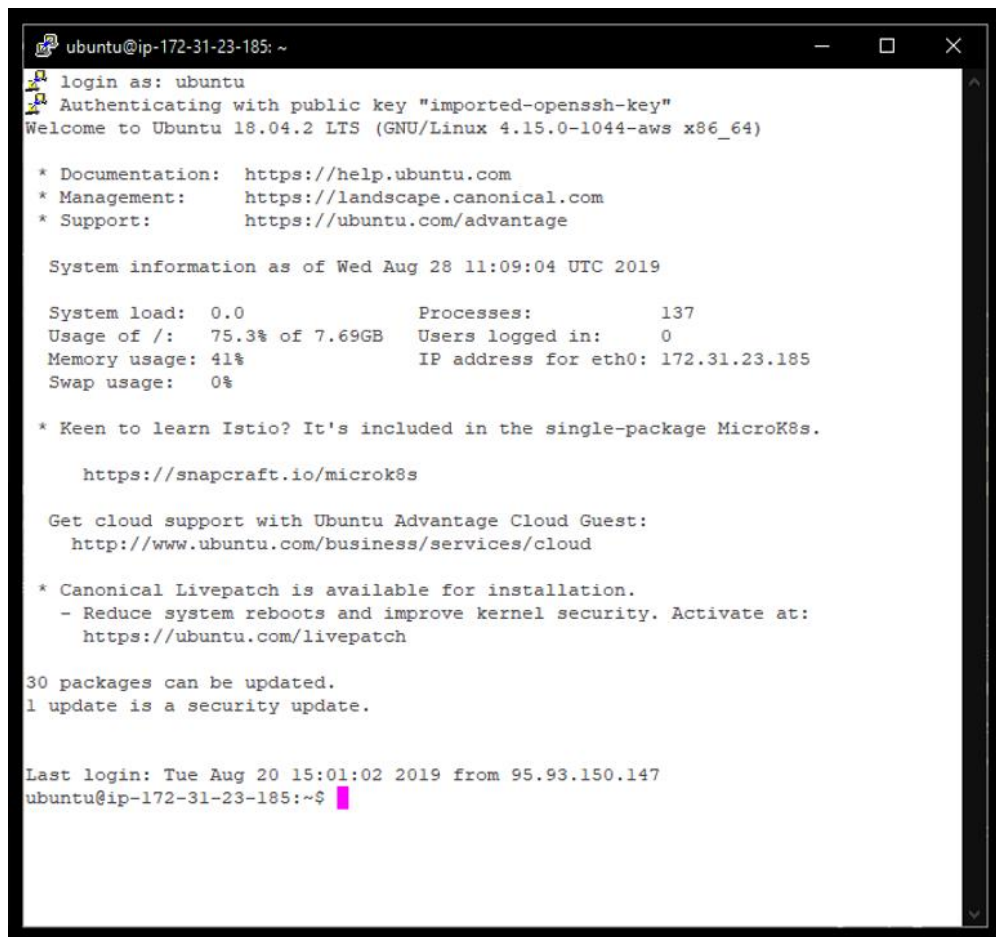


Figura 7.2.2 - Putty, criar sessão e colocar DNS *publico* da instância

A figura 7.2.3, representa o que sucede ao entrarmos no terminal da instância através do Putty. Numa primeira parte é necessário colocar o nome padrão definido pela AWS (tabela 7.1.1) corresponde com o sistema operativo que foi escolhido para a instância, por fim o Putty conecta se à instância via SSH e podemos então trabalhar a instância.



```
ubuntu@ip-172-31-23-185: ~
login as: ubuntu
Authenticating with public key "imported-openssh-key"
Welcome to Ubuntu 18.04.2 LTS (GNU/Linux 4.15.0-1044-aws x86_64)

* Documentation:  https://help.ubuntu.com
* Management:    https://landscape.canonical.com
* Support:       https://ubuntu.com/advantage

System information as of Wed Aug 28 11:09:04 UTC 2019

System load:  0.0           Processes:      137
Usage of /:   75.3% of 7.69GB Users logged in:  0
Memory usage: 41%          IP address for eth0: 172.31.23.185
Swap usage:   0%

* Keen to learn Istio? It's included in the single-package MicroK8s.

https://snapcraft.io/microk8s

Get cloud support with Ubuntu Advantage Cloud Guest:
http://www.ubuntu.com/business/services/cloud

* Canonical Livepatch is available for installation.
- Reduce system reboots and improve kernel security. Activate at:
https://ubuntu.com/livepatch

30 packages can be updated.
1 update is a security update.

Last login: Tue Aug 20 15:01:02 2019 from 95.93.150.147
ubuntu@ip-172-31-23-185:~$
```

Figura 7.2.3 - Putty, Ligação ao terminal da instância vi Putty.

A partir daqui este terminal será igual a qualquer outro terminal da máquina virtual que escolheu, inicialmente a instância irá querer uma actualização ao seu sistema operativo acabando a actualização poderá instalar qualquer programa via linha de comandos na instância e após terminar basta fechar o Putty que a sessão irá suspender também.

7.3 FARSITE

A figura 7.3.1 em baixo representa uma linha do ficheiro RunPanther, que o utilizador terá de escrever. Onde indica quais os ficheiros de entrada que quer utilizar para a simulação e indicar qual o nome com que o conjunto de saídas tenha.

```
/home/ubuntu/examples/Panther/cust/input/local_lcp_1.lcp  
/home/ubuntu/examples/Panther/cust/input/input_3.input  
/home/ubuntu/examples/Panther/cust/input/ignit2.shp 0  
/home/ubuntu/examples/Panther/cust/output/output_runpantherVar 0
```

Figura 7.3.1 - Formato do ficheiro RunPanther.

A figura 7.3.2 em baixo é a linha de comando que o utilizador terá de executar para correr o FARSITE manualmente. Esta linha de comando tem de indicar o caminho para a source do FARSITE bem como o caminho para o ficheiro RunPanther

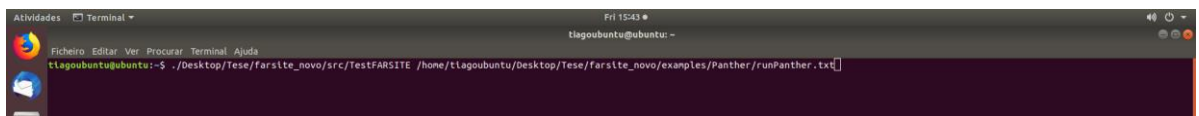


Figura 7.3.2 - Linha de comando para correr o simulador.